Calhoun: The NPS Institutional Archive

DSpace Repository

Theses and Dissertations 1. Thesis and Dissertation Collection, all items

1971-12

On realization of terminal capacity matrices.

Karan, Tahsin

Monterey, California ; Naval Postgraduate School

http://hdl.handle.net/10945/15836

Copyright is reserved by the copyright owner

Downloaded from NPS Archive: Calhoun

Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

http://www.nps.edu/library

ON REALIZATION OF TERMINAL CAPACITY MATRICES

Tahsin Karan

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

ON REALIZATION OF TERMINAL CAPACITY MATRICES

by

Tahsin Karan

Thesis Advisor: S. G. Chan

December 1971

Approved for public release; distribution unlimited.

On Realization of Terminal
Capacity Matrices

by

Tahsin Karan
Lieutenant, Türkish Navy
B.S., Naval Postgraduate School, 1971

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 1971

Thesis

ABSTRACT

This paper presents three algorithms for minimum cost synthesis of an oriented communication net. The realization technique is developed using the min-cut max-flow theorem. The algorithms are able to handle higher order terminal capacities compared to previous methods. Necessary and sufficient conditions are given for the application of the algorithms, which are suitable for computer implementation.

TABLE OF CONTENTS

I. INTRODUCTION - 5

II. THEORETICAL DEVELOPMENT - - - - - - - - - - - - - - - - - - 6

A. PROPERTIES OF TERMINAL CAPACITY MATRIX - - - - - - - - 6

B. THE MAX-FLOW MIN-CUT THEOREM - - - - - - - - - - - - - - 7

C. SYNTHESIS OF NONORIENTED COMMUNICATION NET - - - - - - 8

1. Method of Elementary Matrices - - - - - - - - - - 12

2. Method of Successive Expansion - - - - - - - - - - 15

3. Decomposition of Terminal Capacity Matrices- - - 17

III. SYNTHESIS OF ORIENTED COMMUNICATION NET - - - - - - - - - 20

A. SYNTHESIS OF TERMINAL CAPACITY MATRIX IN THREE-NODE CASE - 20

B. SYNTHESIS OF TERMINAL CAPACITY MATRIX IN FOUR NODE-CASE - 24

1. Algorithm A - 26

2. Algorithm B - 34

3. Algorithm C - 41

4. Dominant Submatrix Partitioning of T Matrices- - 44

5. Flow Chart for Computer Programming - - - - - - 48

IV. CONCLUSION - 49

REFERENCES - 50

INITIAL DISTRIBUTION LIST - - - - - - - - - - - - - - - - - - - 51

FORM DD 1473 - 52

ACKNOWLEDGEMENT

The author wishes to express his appreciation to Professor Shu-Gar Chan for the invaluable aid and councel which he has offered during the preparation of this thesis.

I. INTRODUCTION

The application of the theory of graphs to the analysis of communication nets is natural in the sense that one may consider the various stations of a communication net as vertices and the channels of communication net as branches (lines drawn between these vertices). Every branch has associated with it a nonnegative number called the branch capacity which indicates the maximum amount of information that can pass through the branch. A communication net must have large enough branch capacities such that all message requirements can reach their destinations simultaneously.

In many practical applications, the maximum allowable communication from station i to station j and the maximum allowable communication from station j to station i may be different. For representing such a system, oriented branches must be used, resulting in an oriented graph. Therefore, the branch capacity matrix and the terminal capacity matrix become assymmetrical.

The purpose of this paper is to investigate a synthesis method for oriented communication nets. The necessary conditions and a realization method for up to three-by-three matrix are given by Tang and Chien [2]. The necessary conditions and realization methods for four-by-four matrix are presented in this paper. These ideas may easily be extended to higher-order cases. The method given here is based on the max-flow min-cut theorem [7] and can be adapted for computer solution. Related flow chart for computer programing will be given later in this paper.

II. THEORETICAL DEVELOPMENT

Several authors have worked on communication nets and terminal capacity matrices. Methods for the synthesis of oriented or nonoriented communication nets are given in references [2, 3, 5, 6, 7, 8, 10, 12]. Properties of the terminal capacity matrix, the max-flow min-cut theorem and several methods for analyzing communication nets are presented in this section.

A. PROPERTIES OF TERMINAL CAPACITY MATRIX

1. Oriented Communication Net

The terminal capacity matrix is always partitionable into submatrices and submatrices on the diagonal are again partitionable until each submatrix becomes a one-by-one matrix.

THEOREM 1 [1]. Partitioning of a terminal capacity matrix, if t_1 corresponds to a minimum cut S_1 cutting all directed paths from subgraph A to subgraph B, and if t_2 corresponds to another minimum cut S_2 cutting all directed paths from A_1 to A_2 (both subgraph of A), then S_2 cannot be a minimum cut of any two subgraphs of B unless $t_2=t_1$ and if S_2 is also a minimum cut cutting all directed paths from some B_3 to B_4 (both nonempty subgraphs of B), then there exist at least two more cuts with the same minimum value t.

THEOREM 2 [1]. Let $t_{ij}(i,j=1, 2, \ldots\ldots, n, i\neq j)$ be any element of a terminal capacity matrix; then

$$t_{ij} \geq \min(t_{ik}, t_{kj}) \qquad (2\text{-}1)$$

and $i,j,k=1,2,\ldots\ldots, n,\ i\neq j$

THEOREM 3 [1]. T' is the terminal capacity matrix of graph G' and T'' is the terminal capacity matrix of graph G''. Let,

$$G = G' + G'' \text{ (in terms of edge matrices)} \tag{2-2}$$

and

$$T = T' + T'' \tag{2-3}$$

Then T is the terminal capacity matrix of graph G if and only if for each ordered node pair i and j there exists a cut for all three graphs G, G' and G''.

2. Nonoriented Communication Net

A terminal capacity matrix of a communication net is always partitionable into the submatrices as in oriented case.

The maximal-flow capacity from node i to node j is equal to the maximal-flow capacity from node j to node i.

CORROLLARY 1 [1]. Let S be the minimum cut-set which separates graph G into subgraph G' and G'', the terminal-capacity t is not changed when all edges in G'' are shorted, provided that i andj are both in G'.

B. THE MAX-FLOW MIN-CUT THEOREM

The max-flow min-cut theorem is formulated by Ford and Fulkerson [7]. It can be used to obtain maximum flow in a network.

THEOREM 4 [7]. For an oriented network the maximal-flow from node n_1 to node n_2 is equal to the minimum cut, which cuts all directed paths from n_1 to n_2.

For finding maximal flow of an oriented network, the following procedure may be used with the aid of the theorem given above.

a) Select a pair of vertices. Determine a path such that all forward edges are not saturated ($f<c$) and all reverse edges have nonzero flow. Repeat it if $f = c$.

b) Let Δf_1 be the minimum of all the differences $(c-f)$ for forward edges and Δf_2 be the minimum of all the differences for reverse edges. Increase the flow of the forward edges by an amount $\Delta f = \min(\Delta f_1, \Delta f_2)$, and decrease the flow of reverse edges by an amount Δf.

c) Repeat (a) and (b) until no more paths exist as described in step (a).

C. SYNTHESIS OF NONORIENTED COMMUNICATION NETS

Several authors have investigated methods for realizing nonoriented communication nets. In this section, the method of Mayeda [3], Wing and Chien [5], Gomory and Hu [6] will be briefly presented.

Mayeda's method is based on the realization of communication nets using a branch capacity matrix which is obtained from a terminal capacity matrix. The realization is accomplished by partitioning the terminal capacity and the branch capacity matrices properly.

Suppose that the terminal capacity matrix of a communication net is partitioned as

$$T = \left[\begin{array}{c|c} T_{a1} & T_{(t1)} \\ \hline T'_{(t1)} & T_1 \end{array}\right] \tag{2-4}$$

Let N_1 and N_{a1} be the subnets corresponding to T_1 and T_{a1}, respectively.

Partition the branch capacity matrix in the following form

$$C = \left[\begin{array}{c|c} C_{a1} & C_{(t1)} \\ \hline C'_{(t1)} & C_1 \end{array}\right] \tag{2-5}$$

It can be seen that the rows and the columns of C_1, C_{a1}, and $C_{(t_1)}$ have the same arrangement as the rows and the columns of T_1, T_{a1}, and $T_{(t_1)}$, respectively, the branches whose branch capacities appear in $C_{(t_1)}$ are those which are connected between any vertex in N_1 and any vertex in N_{a1}. Then, t_1 in $T_{(t_1)}$ is equal to the sum of all elements (branch capacities) in $C_{(t_1)}$.

$$t_1 = \text{Sum of all elements in } C_{(t_1)} \tag{2-6}$$

Let a principal partitioning process be applied to the resultant submatrix T_{a1} in Eq. (2-4).

$$T = \left[\begin{array}{cc|c} T_{a2} & T_{(t_2)} & \\ \cline{1-2} T'_{(t_2)} & T_2 & T_{(t_1)} \\ \hline \multicolumn{2}{c|}{T'_{(t1)}} & T_1 \end{array}\right] \tag{2-7}$$

Let the branch capacity matrix in Eq. (2-5) is partitioned as

$$C = \left[\begin{array}{c|c|c} C_{a2} & C_{(t2)} & C_{(t_1)_{a2}} \\ \hline C'_{(t2)} & C_2 & C_{(t1)_2} \\ \hline C'_{(t1)_{a2}} & C'_{(t1)_2} & C_1 \end{array}\right] \tag{2-8}$$

where the rows and columns of C_{a2}, C_2, and $C_{(t2)}$ in (2-8) have the same arrangement as the rows and columns of T_{a2}, T_2, and $T_{(t_2)}$ in (2-7), respectively.

t_2 is equal to any element in $T_{(t_2)}$ and can be written as

$$t_2 = \text{Sum of elements in } C_{(t_2)} + \min\left(\text{Sum of elements in } C_{(t_1)_{a2}},\ \text{Sum of elements in } C_{(t_1)_2}\right) \tag{2-9}$$

Let $V(C_k)$ be the sum of all elements in the submatrix C_k. Then (2-9) can be expressed as

$$t_2 = V(C_{(t_2)}) + \min \left\{ V(C_{(t_1)_{a2}}),\ V(C_{(t_1)_2}) \right\} \tag{2-10}$$

Suppose a principle partitioning process is applied to T in (2-7).

$$T = \left[\begin{array}{cc:c:c} T_{a3} & T_{(t_3)} & & \\ \hdashline T'_{(t_3)} & T_3 & T_{(t_2)} & T_{(t_1)} \\ \hdashline \multicolumn{2}{c:}{T'_{(t_2)}} & T_2 & \\ \hdashline \multicolumn{3}{c:}{T'_{(t_1)}} & T_1 \end{array}\right] \tag{2-11}$$

Let the branch capacity matrix in (2-8) be partitioned as

$$C = \left[\begin{array}{c:c:c:c} C_{a3} & C_{(t_3)} & C_{(t_2)a3} & C_{(t_1)a3} \\ \hdashline C'_{(t_3)} & C_3 & C_{(t_2)3} & C_{(t_1)3} \\ \hdashline C'_{(t_2)a3} & C'_{(t_2)3} & C_2 & C_{(t_1)2} \\ \hdashline C'_{(t_1)a3} & C'_{(t_1)3} & C'_{(t_1)2} & C_1 \end{array}\right] \tag{2-12}$$

Let N_1, N_2, N_3 and N_{a3} be the subnets consisting of the vertices associated to the rows (and the columns) of T_1, T_2, T_3, and T_{a3} respectively. Also let $N(S_3)_1$ and $N(S_3)_2$ be the subnets obtained from the net N by removing every branch in the corresponding cut set S_3 of t_3 where $N(S_3)_1$ contains N_{a3} and $N(S_3)_2$ contains N_3. The subnets N_1 and N_2 can be in either $N(S_3)_1$ or $N(S_3)_2$. Hence $N(S_3)_1$ and $N(S_3)_2$ is the one of the following four subnets.

1) $N(S_3)_1$ contains N_{a3} and $N(S_3)_2$ contains N_1, N_2 and N_3.
2) $N(S_3)_1$ contains N_{a3} and N_1, and $N(S_3)_2$ contains N_2 and N_3.

3) $N(S_3)_1$ contains N_{a3}, N_2, and $N(S_3)_2$ contains N_1 and N_3.

4) $N(S_3)_1$ contains N_{a3}, N_1 and N_2, and $N(S_3)_2$ contains N_3 only.

Thus the corresponding cutset S_3 of t_3 is one of the following four cutsets.

CASE 1) S_a consists of the branches which are connected between any vertex in N_{a3} and any vertex in one of N_1, N_2 and N_3.

CASE 2) S_b consists of the branches which are connected between any vertex in either N_{a3} or N_1 and any vertex in either N_2 or N_3.

CASE 3) S_c consists of the branches which are connected between any vertex in either N_{a3} or N_2 and any vertex in either N_1 or N_3.

CASE 4) S_d consists of the branches which are connected between any vertex in any one of N_{a3}, N_1 and N_2 and any vertex in N_3.

The branch capacities of the branches, which are connected between any vertex in $N(S_3)_1$ and any vertex in $N(S_3)_2$, are the elements in C at the intersection of Set-1 and Set-2. Set-1 is the rows representing the vertices of $N(S_3)_1$ and Set-2 is the columns representing the vertices of $N(S_3)_2$. Therefore, the cutset S_a [mentioned above in CASE 1] is the set of elements of C which are the intersections of the rows of C_{a3} (representing the vertices in N_{a3}) and the columns of C_1, C_2 and C_3 (representing the vertices in N_1, N_2, and N_3) in (2-12).

$$V(S_a) = V(C_{(t_3)}) + V(C_{(t_2)a2} + V(C_{(t_1)a3}) \qquad (2\text{-}13)$$

The cutset S_b [mentioned above in CASE 2] is the set of elements of C which are the intersections of the rows of C_{a3}, and C_1 (representing the vertices in N_{a3} and N_1) and the columns of C_2 and C_3 (representing the vertices in N_2 and N_3) in (2-12).

$$V(S_b) = V(C_{(t_3)}) + V(C_{(t_2)a3}) + V(C'_{(t_1)_3}) + V(C'_{(t_1)_2}) \qquad (2\text{-}14)$$

Likewise the values of S_c and S_d (mentioned above in CASE 2 and CASE 3) are

$$V(S_c) = V(C_{(t_3)}) + V(C_{(t_1)a3}) + V(C'_{(t_2)_3}) + V(C_{(t_1)_2}) \qquad (2\text{-}15)$$

$$V(S_d) = V(C_{(t_3)}) + V(C'_{(t_2)_3}) + V(C'_{(t_1)_3}) \qquad (2\text{-}16)$$

t_3 is the minimum value of $V(S_a)$, $V(S_b)$, $V(S_c)$ and $V(S_d)$, t_3 is equal to

$$\begin{aligned} t_3 = V(C_{(t_3)}) + \min \Big\{ & V(C_{(t_2)a3}) + V(C_{(t_1)a3}),\ V(C_{(t_2)a3}) + V(C'_{(t_1)_3}) \\ & + V(C'_{(t_1)_2}),\ V(C_{(t_1)a3}) + V(C'_{(t_2)_3}) + V(C_{(t_1)_2}),\ V(C'_{(t_2)_3}) \\ & + V(C'_{(t_1)_3}) \Big\} \end{aligned} \qquad (2\text{-}17)$$

From (2-17) $V(C_{(t_3)})$ can be find and also the subgraph N_3 can be formed.

We can apply the principal partitioning process to T_{a3} in (2-11), and by continuing the same procedure the branch capacity matrix can be obtained. The number of the steps depends on the order of T.

1. Method of Elementary Matrices

The method of elementary matrices [1], [5] requires a maximum of $\frac{1}{2}n(n-1)$ branches, where n is the number of nodes. Elementary terminal capacity matrix can be put into the following form

$$T = \begin{bmatrix} d & t_1 & t_2 & t_3 & \cdot & \cdot & \cdot & \cdot & t_{n-1} \\ t_1 & d & t_2 & t_3 & \cdot & \cdot & \cdot & \cdot & t_{n-1} \\ t_2 & t_2 & d & t_3 & \cdot & \cdot & \cdot & \cdot & t_{n-1} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ t_{n-1} & t_{n-1} & t_{n-1} & t_{n-1} & \cdot & \cdot & \cdot & \cdot & d \end{bmatrix} \qquad (2\text{-}18)$$

where $t_1 \geq t_2 \geq t_3 \geq \ldots\ldots\ldots\ldots \geq t_{n-1}$

Every elementary terminal capacity matrix is guaranteed to be realizable [3].

Figure 2-1 realizes an elementary terminal capacity matrix of order n with minimum total edge-capacity.

If a terminal capacity matrix T of order n is pártitionable as

$$T = \left[\begin{array}{c:c} T_1 & \begin{matrix} t_o & t_o & \cdots & t_o \\ t_o & t_o & \cdots & t_o \\ \vdots & \vdots & & \vdots \\ t_o & t_o & \cdots & t_o \end{matrix} \\ \hdashline \begin{matrix} t_o & t_o & \cdots & t_o \\ t_o & t_o & \cdots & t_o \\ \vdots & \vdots & & \vdots \\ t_o & t_o & \cdots & t_o \end{matrix} & T_2 \end{array}\right] \qquad (2\text{-}19)$$

where $t_o = \min_{i,j} (t_{ij})$ and T_1 and T_2 are elementary terminal-capacity matrices of order k and n-k respectively, T can be realized by a net as shown in Fig. 2-2. The two "linking" branches a and b can be placed between any two pairs of nodes. If the T matrix is partitionable into $T_1, T_2, \ldots, T_p$ elementary terminal-capacity matrices, realization is shown in Fig. 2-3. The number of branches required for this realization is at most 2n-p-2, where p is the number of elementary terminal-capacity matrices in a given T.

Example 1. The realization of following terminal capacity matrix is given in Fig. 2-4.

$$T = \left[\begin{array}{ccc:cc:ccc} \textcircled{1} & 14 & 12 & 4 & 4 & 4 & 4 & 4 \\ 14 & \textcircled{2} & 12 & 4 & 4 & 4 & 4 & 4 \\ 12 & 12 & \textcircled{3} & 4 & 4 & 4 & 4 & 4 \\ \hdashline 4 & 4 & 4 & \textcircled{4} & 10 & 6 & 6 & 6 \\ 4 & 4 & 4 & 10 & \textcircled{5} & 6 & 6 & 6 \\ \cdashline{4-8} 4 & 4 & 4 & 6 & 6 & \textcircled{6} & 8 & 6 \\ 4 & 4 & 4 & 6 & 6 & 8 & \textcircled{7} & 6 \\ 4 & 4 & 4 & 6 & 6 & 6 & 6 & \textcircled{8} \end{array}\right] \qquad (2\text{-}20$$

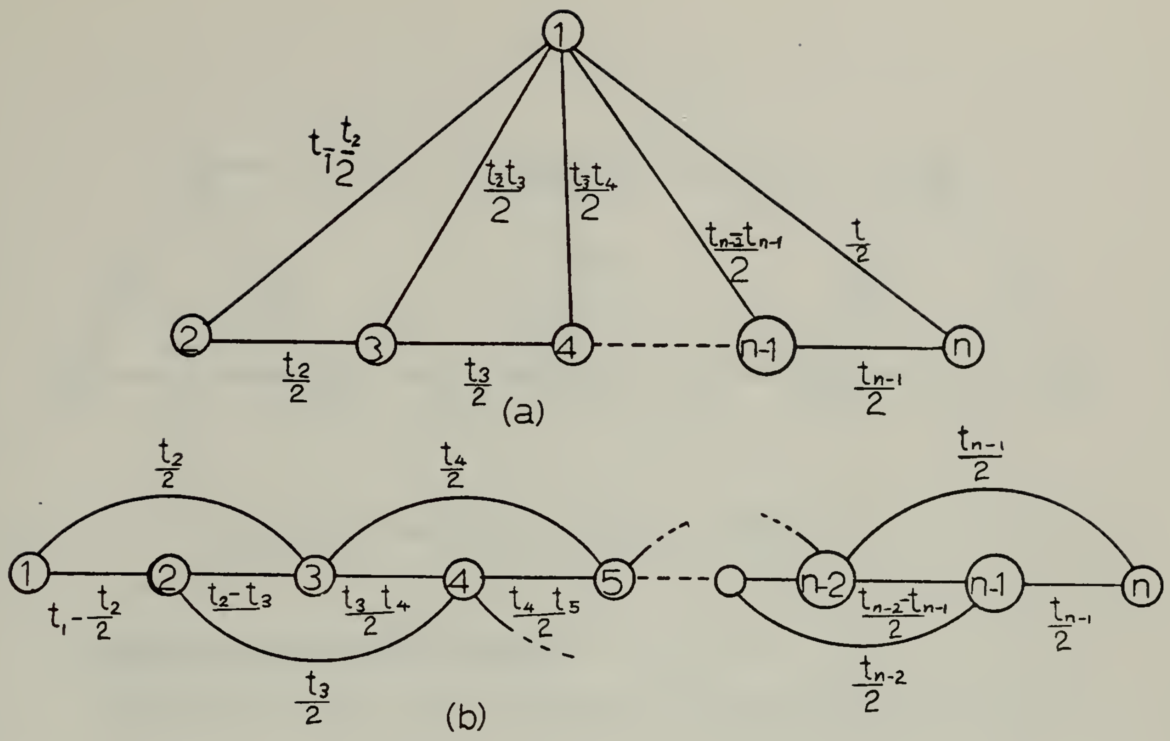


Fig. 2-1 Realization of Elementary Terminal Capacity Matrix

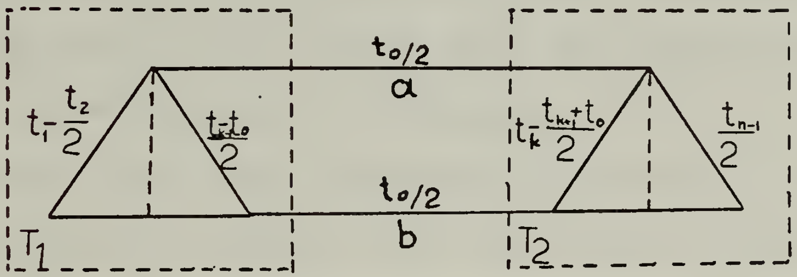


Fig. 2-2 Combination of Elementary Nets

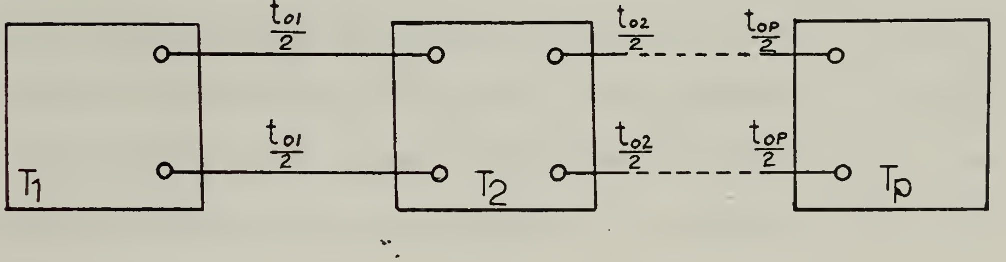


Fig. 2-3 Realization Through Combination of Elementary Nets

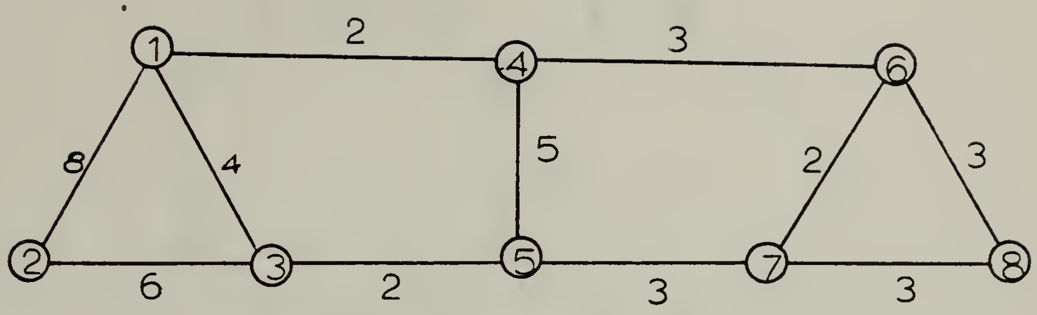


Fig. 2-4 Realization of Eq. (2-20)

2. Method of Successive Expansion

The method of successive expansion [5] employes relatively few edges. The number of edges required by this method is exactly $P'+n-1$, where P' is the index of partitioning and n is the number of nodes. The index of partitioning is the number of operations necessary to partition the matrix T into a form in which every diagonal submatrix is either of order two-by-two or one-by-one.

In this method, first, the diagonal submatrices of the first partition are treated as nodes to form a ring, with each ring element equal to $\frac{1}{2}t_0$, half of the capacity of first partition. To realize each diagonal submatrix, a new ring is formed and each branch in the new ring will have a capacity of $\frac{1}{2}t_1$ except one branch, which has a capacity of $\frac{1}{2}(t_1-t_0)$ and this branch is shared by the new ring and the original ring as shown in Fig. 2-5. Each submatrix is treated as above except one case in which the submatrix is of order two so that the two branches of the new ring may be combined to form one branch.

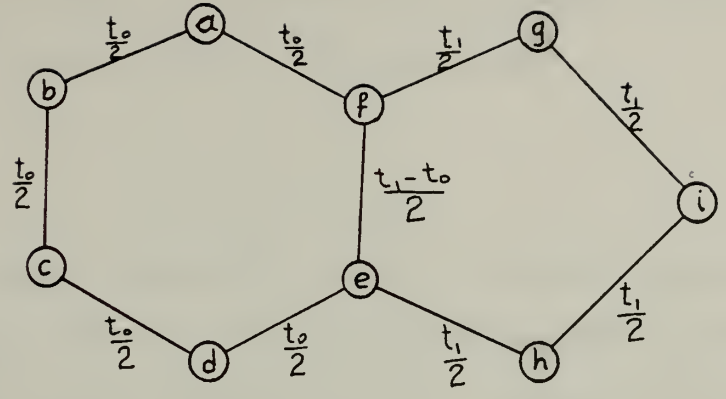
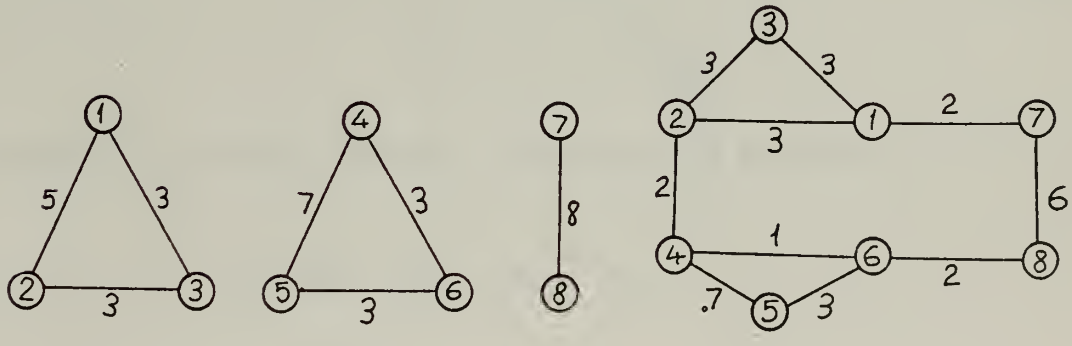


Fig. 2-5 Combination of Ring Structures in the Process of Realization

Example 2. Realization of following terminal capacity matrix is given in Fig. 2-6.

$$T = \left[\begin{array}{ccc|ccc|cc}
\textcircled{1} & 8 & 6 & 4 & 4 & 4 & 4 & 4 \\
8 & \textcircled{2} & 6 & 4 & 4 & 4 & 4 & 4 \\
6 & 6 & \textcircled{3} & 4 & 4 & 4 & 4 & 4 \\
\hline
4 & 4 & 4 & \textcircled{4} & 10 & 6 & 4 & 4 \\
4 & 4 & 4 & 10 & \textcircled{5} & 6 & 4 & 4 \\
4 & 4 & 4 & 6 & 6 & \textcircled{6} & 4 & 4 \\
\hline
4 & 4 & 4 & 4 & 4 & 4 & \textcircled{7} & 8 \\
4 & 4 & 4 & 4 & 4 & 4 & 8 & \textcircled{8}
\end{array}\right] \qquad (2\text{-}21)$$



(a) (b)

Fig. 2-6 a) Realization of diagonal submatrices.
b) Realization of T matrix.

3. Decomposition of Terminal Matrices

With the method of decomposition of terminal matrices [6] the terminal capacity matrix can be written as

$$T = T_1 + T_u \tag{2-22}$$

where t_u is uniform element in T_u and it is equal to the minimum element of $t_{i,j}$. Zero elements of T_1 indicate where the minimum cut-set will be in the realization of T_1. Minimum cut sets of the realization of T_u correspond to the minimum cut-sets of the realization of T_1. T matrix can be written the sum of uniform matrices, as

$$T = \sum T_{u_i} \tag{2-23}$$

The T_{u_i}'s are realized and combined in such a way that all their minimum cut-sets correspond to each other.

Example 3. We shall realize the following T matrix with the method described above.

$$T = \begin{bmatrix} \textcircled{1} & 7 & 6 & 3 & 3 \\ 7 & \textcircled{2} & 6 & 3 & 3 \\ 6 & 6 & \textcircled{3} & 3 & 3 \\ 3 & 3 & 3 & \textcircled{4} & 5 \\ 3 & 3 & 3 & 5 & \textcircled{5} \end{bmatrix} \tag{2-24}$$

T matrix can be written the sum of four uniform matrices, as

$$T = T_{u_1} + T_{u_2} + T_{u_3} + T_{u_4}$$

$$T_{u_1} = \begin{bmatrix} \textcircled{1} & 3 & 3 & 3 & 3 \\ 3 & \textcircled{2} & 3 & 3 & 3 \\ 3 & 3 & \textcircled{3} & 3 & 3 \\ 3 & 3 & 3 & \textcircled{4} & 3 \\ 3 & 3 & 3 & 3 & \textcircled{5} \end{bmatrix} \qquad (2\text{-}25)$$

$$T_{u_2} = \begin{bmatrix} \textcircled{1} & 0 & 0 & 0 & 0 \\ 0 & \textcircled{2} & 0 & 0 & 0 \\ 0 & 0 & \textcircled{3} & 0 & 0 \\ 0 & 0 & 0 & \textcircled{4} & 2 \\ 0 & 0 & 0 & 2 & \textcircled{5} \end{bmatrix} \qquad (2\text{-}26)$$

$$T_{u_3} = \begin{bmatrix} \textcircled{1} & 3 & 3 & 0 & 0 \\ 3 & \textcircled{2} & 3 & 0 & 0 \\ 3 & 3 & \textcircled{3} & 0 & 0 \\ 0 & 0 & 0 & \textcircled{4} & 0 \\ 0 & 0 & 0 & 0 & \textcircled{5} \end{bmatrix} \qquad (2\text{-}27)$$

$$T_{u_4} = \begin{bmatrix} \textcircled{1} & 1 & 0 & 0 & 0 \\ 1 & \textcircled{2} & 0 & 0 & 0 \\ 0 & 0 & \textcircled{3} & 0 & 0 \\ 0 & 0 & 0 & \textcircled{4} & 0 \\ 0 & 0 & 0 & 0 & \textcircled{5} \end{bmatrix} \qquad (2\text{-}28)$$

Realization of each T_{u_i} matrix is given in Fig. 2-7 and realization of the T matrix is given in Fig. 2-8.

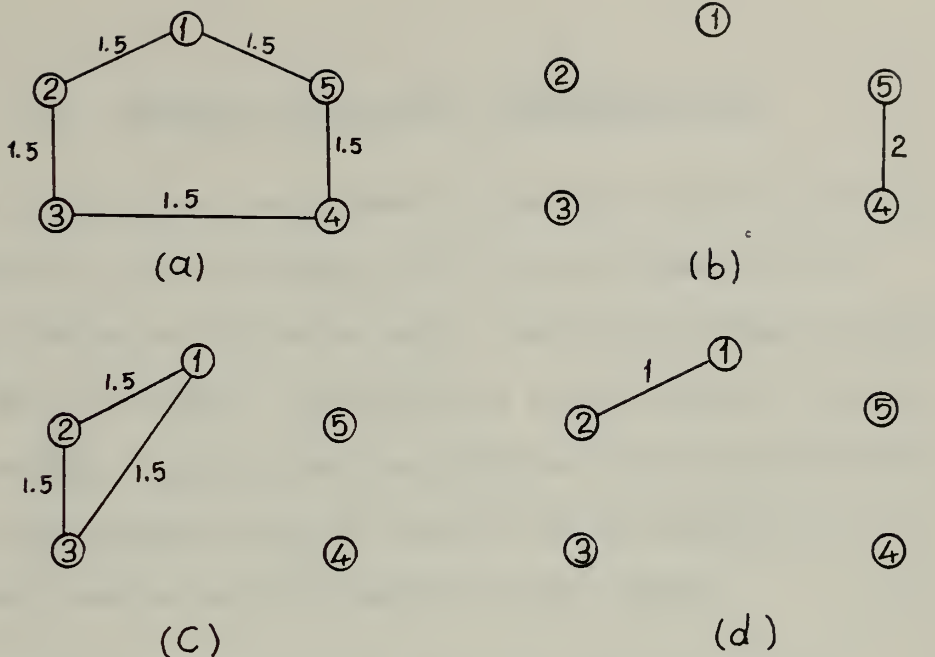


Fig. 2-7 Realization of T_{u_1} in (a), T_{u_2} in (b), T_{u_3} in (c) and T_{u_4} in (d)

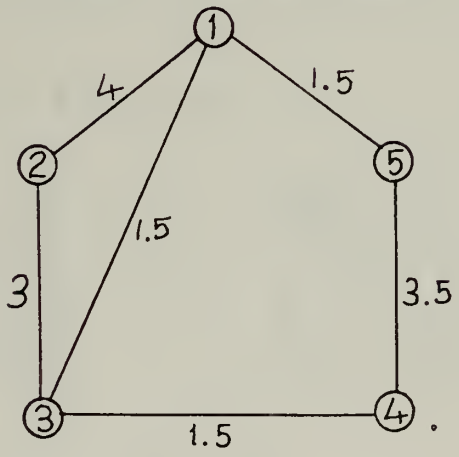


Fig. 2-8. Realization of T matrix

III. SYNTHESIS OF ORIENTED COMMUNICATION NET

Several methods are investigated by Resh [8], Frisch and Sen [9], Tang and Chien [2], Hu and Gomory [12], Chou and Frank [10] for realizing oriented communication nets. The method of Tang and Chien is given in the next section. It applies to a three-by-three terminal capacity matrix. In this paper the technique for the realization of a four-by-four terminal capacity matrices and its extension to higher-order terminal capacity matrices will be given.

A. SYNTHESIS OF TERMINAL CAPACITY MATRIX IN THREE-NODE CASE

The terminal capacity matrix is partitioned as

$$T = \left[\begin{array}{c:c:c} \textcircled{1} & \multicolumn{2}{c}{t_1} \\ \hdashline t_{21} & \textcircled{2} & t_{23} \\ \hdashline t_{31} & t_{32} & \textcircled{3} \end{array}\right] \qquad (3\text{-}1)$$

It can be written as the sum of two matrices, T' and T_c

$$T = T' + T_c$$

$$T' = \left[\begin{array}{c:c:c} \textcircled{1} & \multicolumn{2}{c}{t_1} \\ \hdashline t'_{21} & \textcircled{2} & t'_{23} \\ \hdashline t'_{31} & t'_{32} & \textcircled{3} \end{array}\right] \quad ; \quad T_c = \left[\begin{array}{c:c:c} \textcircled{1} & t_1 & t_1 \\ \hdashline t_1 & \textcircled{2} & t_1 \\ \hdashline t_1 & t_1 & \textcircled{3} \end{array}\right]$$

where $t'_{ij} = t_{ij} - t_1 \geq 0$ for $i \neq j$

To realize T', we first realize the two-by-two submatrix (containing nodes 1 and 2) as in Fig. 3-1a. Since the first row in T' contains zero entires the only connection between the subgraph shown in Fig. 3-1a and node 1 should be from nodes 2 and 3 to node 1 as shown in Fig. 3-1b with branch capacities x and y. In order to realize T' as in Fig. 3-1b the minimum cut requirements for T must be satisfied. We can obtain the following equations.

$$\min\left[(x+y),\ (x+t'_{23})\right] = t'_{21} \qquad (3\text{-}2)$$

$$\min\left[(x+y),\ (y+t'_{32})\right] = t'_{31} \qquad (3\text{-}3)$$

or

$$x = \max\left[(t'_{21}-t'_{23}),(t'_{21}-y)\right] \qquad (3\text{-}4)$$

$$y = \max\left[(t'_{31}-t'_{32}),(t'_{31}-x)\right] \qquad (3\text{-}5)$$

If we represent equations (3-4) and (3-5) on the x-y plane, we obtain two curves as shown in Fig. 3-2. The intersection of these curves is (x_0,y_0) where, if $t'_{31} \geq t'_{21}$,

$$x_0 = t'_{21}-t'_{23} \qquad (3\text{-}6)$$

$$y_0 = \max\left[(t'_{31}-t'_{32}),(t'_{31}-t'_{21}+t'_{23})\right] \qquad (3\text{-}7)$$

if $t'_{31} \geq t'_{21}$,

$$x_0 = \max\left[(t'_{21}-t'_{23}),(t'_{21}-t'_{31}+t'_{32})\right] \qquad (3\text{-}8)$$

$$y_0 = t'_{31}-t'_{32} \qquad (3\text{-}9)$$

The realization of constant matrix T_c is a graph with constant cuts. It is either a cycle oriented in either direction, or two cycles oriented in different directions. Minimum cuts of T, T' and T_c are idential, then conditions are satisfied in Theorem-3 .

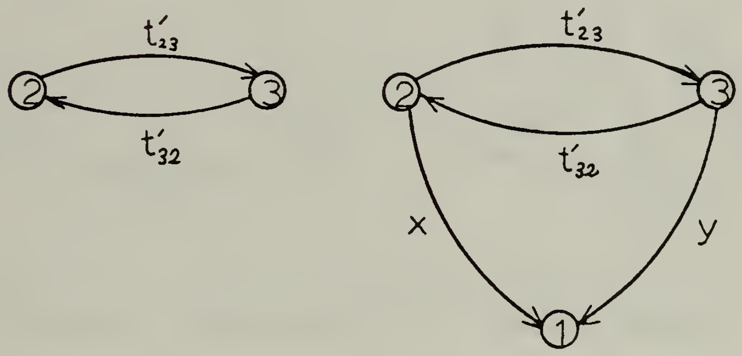


Fig. 3-1. Realization of T'

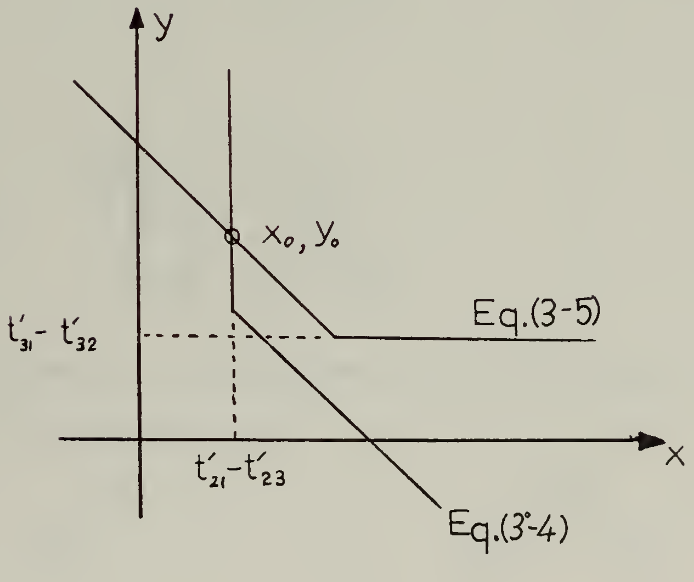


Fig. 3-2. Curves of Equations (3-4) and (3-5)

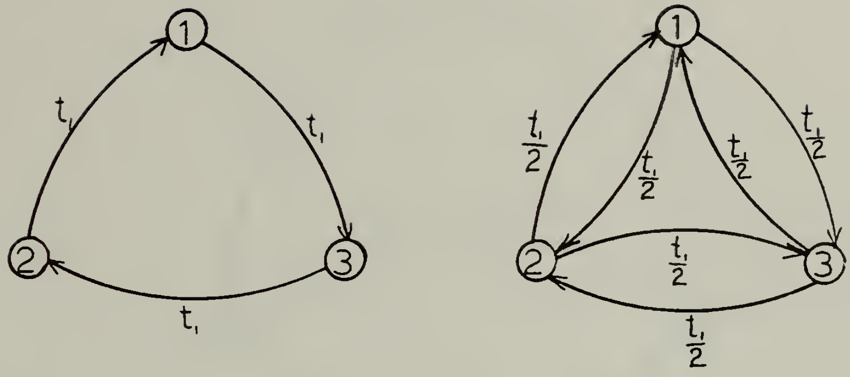


Fig. 3-3. Realization of a Constant Matrix

Example 4. The following T matrix is to be realized:

$$T = \left[\begin{array}{c|cc} \textcircled{1} & 2 & 2 \\ \hline 3 & \textcircled{2} & 3 \\ 3 & 4 & \textcircled{3} \end{array}\right] \qquad (3\text{-}10)$$

We may write:

$$T = T' + T_c = \left[\begin{array}{c|cc} \textcircled{1} & 0 & 0 \\ \hline 1 & \textcircled{2} & 1 \\ 1 & 2 & \textcircled{3} \end{array}\right] + \left[\begin{array}{ccc} \textcircled{1} & 2 & 2 \\ 2 & \textcircled{2} & 2 \\ 2 & 2 & \textcircled{3} \end{array}\right] \qquad (3\text{-}11)$$

The realizations of T' and T_c are in Fig. 3-4a and b. The final realization of T is in Fig. 3-5.

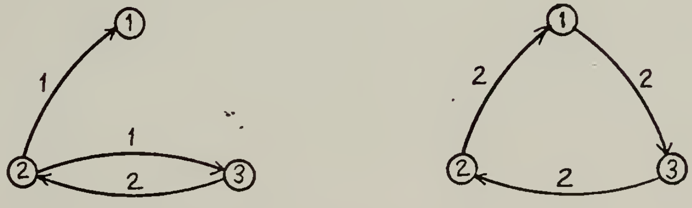


Fig. 3-4. Realizations of T' and T_c matrices

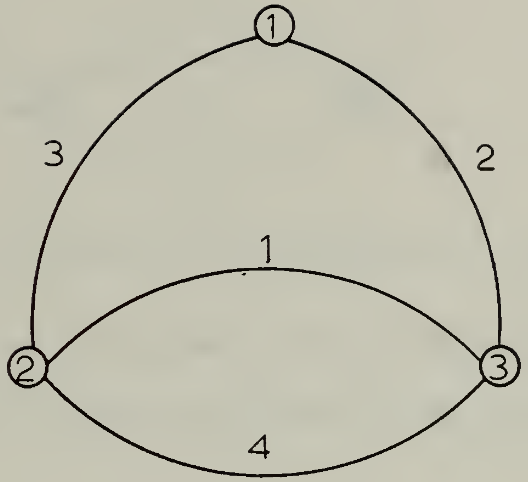


Fig. 3-5. The final realization of T matrix

B. SYSTHESIS OF TERMINAL CAPACITY MATRIX IN FOUR-NODE CASE

The terminal capacity matrix of an oriented communication net containing four nodes can be partitioned as

$$T = \left[\begin{array}{cc|cc} \textcircled{1} & t_{12} & t_1 & t_1 \\ t_{21} & \textcircled{2} & t_1 & t_1 \\ \hline t_{31} & t_{32} & \textcircled{3} & t_{34} \\ t_{41} & t_{42} & t_{43} & \textcircled{4} \end{array}\right] \qquad (3\text{-}12)$$

The form of the realization is given in Fig. 3-6, where $t'_{ij}=t_{ij}-t_1$.

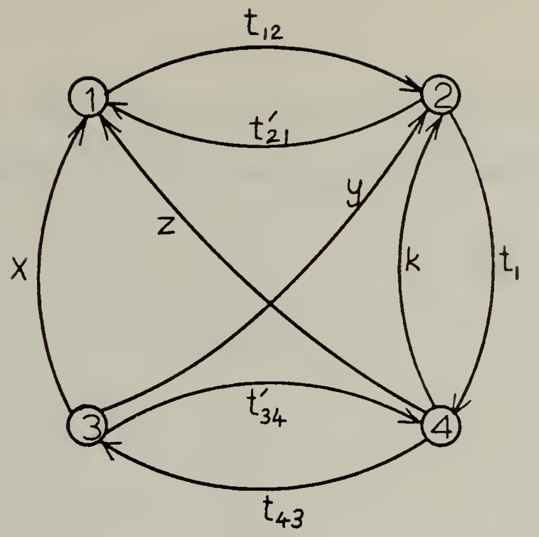


Fig. 3-6. The form of the realization of T matrix

In order to use the algorithms which will be given later in this section, the terminal capacity matrix must be in one of the following three forms.

Form (1) $t_{31} < t_{41}$; $t_{31} < t_{42}$

$t_{32} < t_{41}$; $t_{32} < t_{42}$

Form (2) $t_{31} < t_{32}$; $t_{31} < t_{42}$

$t_{41} < t_{32}$; $t_{41} < t_{42}$

Form (3) $t_{31} < t_{32}$; $t_{31} < t_{41}$

$t_{42} < t_{32}$; $t_{42} < t_{41}$

Form (1) can be realized with Algorithm-A to be given later in section 3-B-1. Form (2) and Form (3) can be realized with Algorithm-B and Algorithm-C to be given later in sections 3-B-2 and 3-B-3, respectively. If the terminal capacity matrix is not in one of these forms, its rows and columns may be rearranged and put into the form of one of them.

The following conditions are necessary for the realization of a four-by-four terminal capacity matrix, using the algorithms given later.

$$t_{31} \geq t'_{21} + t'_{34} + t_1$$

$$t_{32} \geq t_{12} + t'_{34}$$

$$t_{41} \geq t_{43} + t'_{21}$$

$$t_{42} \geq t_{43} + t_{12}$$

1. Algorithm A

1) Determine x and y by writing the following equations, with the aid of Fig. 3-7:

$$t_{31} = \min\left[(x+y+t'_{34}),\ (x+t'_{21}+t'_{34}+t_1)\right] \tag{3-13}$$

$$t_{32} = \min\left[(x+y+t'_{34}),\ (y+t_{12}+t'_{34})\right] \tag{3-14}$$

Which we obtain

$$x = \max\left[(-y+t_{31}-t'_{34}),\ (t_{31}-t'_{21}-t'_{34}-t_1)\right] \tag{3-15}$$

$$y = \max\left[(-x+t_{32}-t'_{34}),\ (t_{32}-t_{12}-t'_{34})\right] \tag{3-16}$$

2) There are several sets of equations for determining z and k, choose proper case, use related equations and apply to Fig. 3-8 for obtaining z and k.

CASE 1: $x+y \leq t_{43}$

$$t_{41} = \min\left[(x+y+z+k),\ (x+z+t'_{21})\right] \tag{3-17}$$

$$t_{42} = \min\left[(x+y+z+k),\ (y+k+t_{12})\right] \tag{3-18}$$

Which we get

$$z = \max\left[(-k+t_{41}-x-y),\ (t_{41}-x-t'_{21})\right] \tag{3-19}$$

$$k = \max\left[(-z+t_{42}-x-y),\ (t_{42}-y-t_{12})\right] \tag{3-20}$$

CASE 2: $x+y > t_{43}$

a. $x \geq t_{43}$

$$t_{41} = \min\left[(z+k+t_{43}),\ (z+t_{43}+t'_{21})\right] \tag{3-21}$$

$$z = \max\left[(-k+t_{41}-t_{43}),\ (t_{41}-t_{43}-t'_{21})\right] \tag{3-22}$$

b. $x < t_{43}$

$$t_{41} = \min\left[(z+k+t_{43}),\ (z+x+t'_{21})\right] \tag{3-23}$$

$$z = \max\left[(-k+t_{41}-t_{43}),\ (t_{41}-x-t'_{21})\right] \tag{3-24}$$

c. $y \geq t_{43}$

$$t_{42} = \min\left[(z+k+t_{43}),\ (k+t_{43}+t_{12})\right] \tag{3-25}$$

$$k = \max\left[(-z+t_{42}-t_{43}),\ (t_{42}-t_{43}-t_{12})\right] \tag{3-26}$$

d. $y < t_{43}$

$$t_{42} = \min\left[(z+k+t_{43}),\ (k+y+t_{12})\right] \tag{3-27}$$

$$k = \max\left[(-z+t_{42}-t_{43}),\ (t_{42}-y-t_{12})\right] \tag{3-28}$$

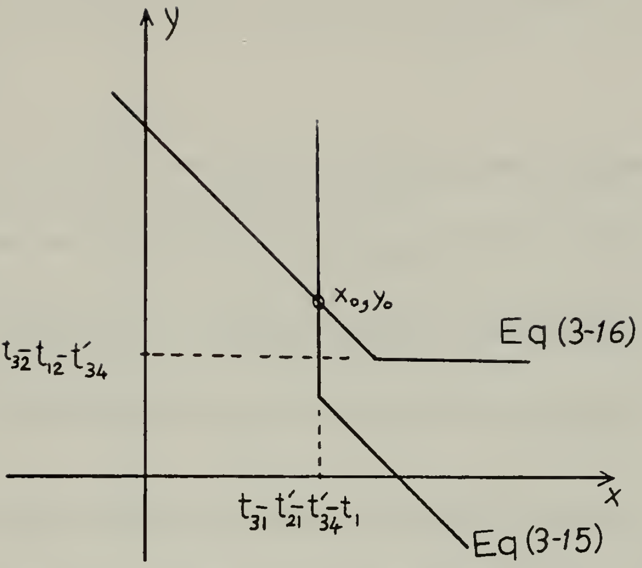


Fig. 3-7. "Curves" of (3-15) and (3-16)

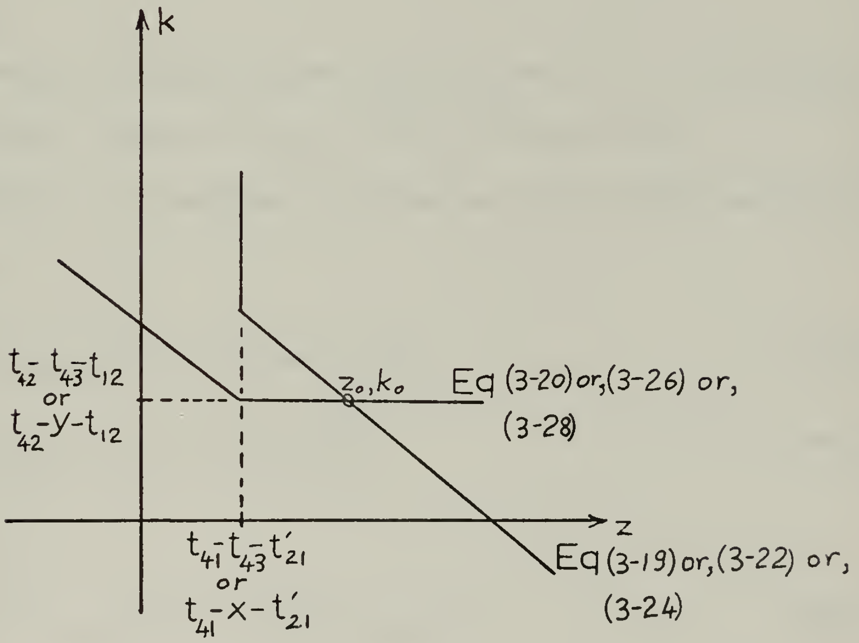


Fig. 3-8. "Curves" of (3-19) or (3-22) or (3-24) and (3-20) or (3-26) or (3-28)

Proof of Algorithm-A: Assume the form of realization as given in Fig. 3-6. Using the max-flow min-cut theorem the following equations can be written.

$$t_{31} = \min\left[(x+y+z+k),(x+z+t'_{21}),(x+y+t'_{34}),(x+t'_{21}+t'_{34}+t_1)\right] \qquad (3\text{-}29)$$

$$t_{32} = \min\left[(x+y+z+k),(y+k+t_{12}),(x+y+t'_{34}),(y+t_{12}+t'_{34})\right] \qquad (3\text{-}30)$$

$$t_{41} = \min\left[(x+y+z+k),(z+k+t_{43}),(x+z+t'_{21}),(z+t_{43}+t'_{21})\right] \qquad (3\text{-}31)$$

$$t_{42} = \min\left[(x+y+z+k),(z+k+t_{43}),(y+k+t_{12}),(k+t_{43}+t_{12})\right] \qquad (3\text{-}32)$$

Algorithm-A applies under the following cases, $t_{31} < t_{41}$ and t_{42}, $t_{32} < t_{41}$ and t_{42}. Then, from equations (3-29) through (3-32) we can state that

(a) x+y+z+k can not be min-cut for t_{31} and t_{32} because it is in t_{41} and t_{42}.

(b) $x+z+t'_{21}$ and $y+k+t_{12}$ can not be min-cut for t_{31} and t_{32} respectively because they are in t_{41} and t_{42} respectively.

Then, the following two equations are obtained from (3-29) and (3-30) for t_{31} and t_{32}:

$$t_{31} = \min\left[(x+y+t'_{34}),(x+t'_{21}+t'_{34}+t_1)\right] \qquad (3\text{-}33)$$

$$t_{32} = \min\left[(x+y+t'_{34}),(y+t_{12}+t'_{34})\right] \qquad (3\text{-}34)$$

We obtain

$$x = \max\left[(-y+t_{31}-t'_{34}),(t_{31}-t'_{21}-t'_{34}-t_1)\right] \qquad (3\text{-}35)$$

$$y = \max\left[(-x+t_{32}-t'_{34}),(t_{32}-t_{12}-t'_{34})\right] \qquad (3\text{-}36)$$

We can solve the above two equations for x and y or we may use a graphical solution for obtaining x and y.

For the determination of z and k, there are two cases to be considered; (1) $x+y \le t_{43}$ and (2) $x+y > t_{43}$. These cases will give different sets of equations.

Case-1) when $x+y \le t_{43}$, automatically both x and $y \le t_{43}$. So $z+x+t'_{21} \le z+t_{43}+t'_{21}$, then $z+t_{43}+t'_{21}$ is eliminated from (3-31). Also $x+y+z+k \le z+k+t_{43}$ because $t_{43} \ge x+y$, then $z+k+t_{43}$ is eliminated from (3-31) because it can not be min-cut for t_{41}. Thus, (3-31) becomes

$$t_{41} = \min\left[(x+y+z+k),(x+z+t'_{21})\right] \qquad (3\text{-}37)$$

We can apply the same logic to the equation (3-32). $x+y \le t_{43}$ so $y \le t_{43}$, then $z+k+t_{43}$ and $k+t_{43}+t_{12}$ are eliminated from (3-32), where (3-32) becomes

$$t_{42} = \min\left[(x+y+z+k),(y+k+t_{12})\right] \qquad (3\text{-}38)$$

from which we get

$$z = \max\left[(-k+t_{41}-x-y),(t_{41}-x-t'_{21})\right] \qquad (3\text{-}39)$$

$$k = \max\left[(-z+t_{42}-x-y),(t_{42}-y-t_{12})\right] \qquad (3\text{-}40)$$

The solution for x and y is found as before. In order to obtain z and k we can use a graphical solution using (3-39) and (3-40).

Case 2) When $x+y > t_{43}$, we can eliminate $x+y+z+k$ from (3-31) and (3-32) because $x+y+z+k > Z+k+t_{43}$ then it can not be min-cut for t_{41} and t_{42}. Thus equations (3-31) and (3-32) become

$$t_{41} = \min\left[(z+k+t_{43}),(z+t_{43}+t'_{21}),(x+z+t'_{21})\right] \qquad (3\text{-}41)$$

$$t_{42} = \min\left[(z+k+t_{43}),(k+t_{43}+t_{12}),(k+y+t_{12})\right] \qquad (3\text{-}42)$$

In this case $x \gtreqless t_{43}$ and $y \gtreqless t_{43}$, and the right-hand sides of equations (3-41) and (3-42) are affected as follows:

$x \geq t_{43}$ eliminates $x+z+t'_{21}$ from (3-41)

$x < t_{43}$ eliminates $z+t_{43}+t'_{21}$ from (3-41)

$y \geq t_{43}$ eliminates $k+y+t_{12}$ from (3-42)

$y < t_{43}$ eliminates $k+t_{43}+t_{12}$ from (3-42)

and each case will give a set of equations for determining z and k.

Example 5. We shall realize the following terminal capacity matrix with using Algorithm-A.

$$T = \left[\begin{array}{cc|cc} \textcircled{1} & 6 & 3 & 3 \\ 5 & \textcircled{2} & 3 & 3 \\ \hline 10 & 11 & \textcircled{3} & 6 \\ 12 & 13 & 7 & \textcircled{4} \end{array}\right] \tag{3-43}$$

1) Applying the numerical values to (3-15) and (3-16) the following equations are obtained:

$$x = \max\left[(-y+7),(2)\right] \tag{3-44}$$

$$y = \max\left[(-x+8),(2)\right] \tag{3-45}$$

With the above equations and using Fig. 3-9, x=2 and y=6 are obtained.

2) $x+y=8 > t_{43}=7$ then Case-2, also $x < t_{43}=7$ then Case 2b and $y < t_{43}=7$ then Case 2d. Applying the numerical values to (3-24) and (3-28) the following equations are obtained:

$$z = \max\left[(-k+5),(8)\right] \tag{3-46}$$

$$k = \max\left[(-z+6),(1)\right] \tag{3-47}$$

With above equations and using Fig. 3-10, $z=8$ and $k=1$ are obtained.

Realization of T is given in Fig. 3-11.

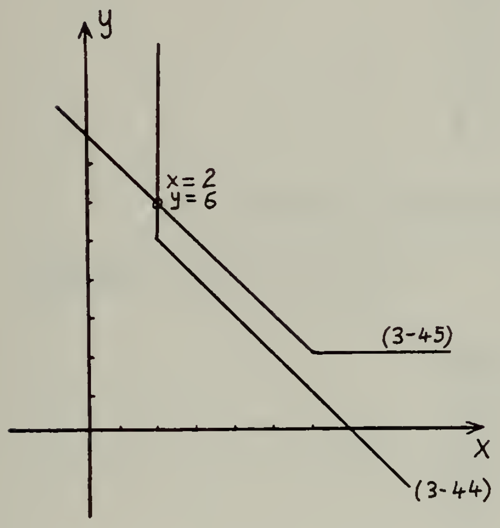


Fig. 3-9

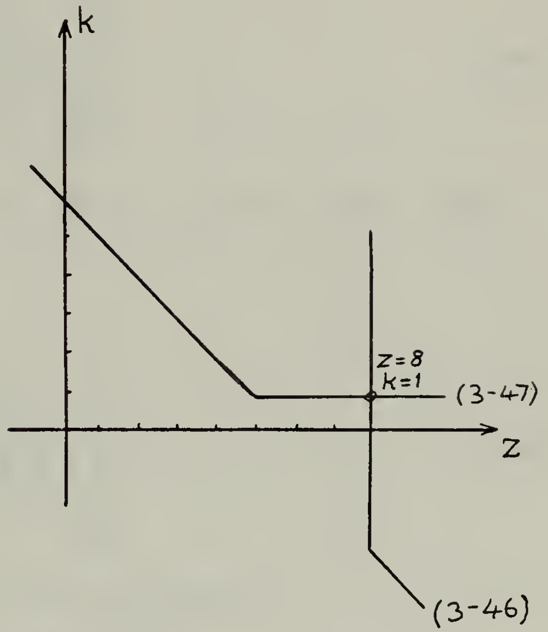


Fig. 3-10

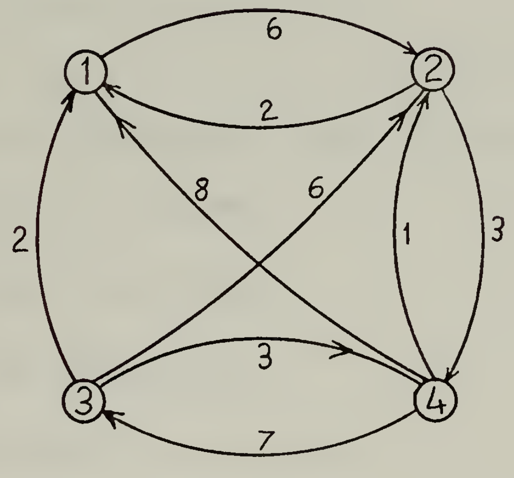


Fig. 3-11. Realization of T matrix in (3-43)

Example 6. We shall realize the following terminal capacity matrix.

$$T = \left[\begin{array}{cc:cc} \textcircled{1} & 5 & 2 & 2 \\ 4 & \textcircled{2} & 2 & 2 \\ \hdashline 9 & 11 & \textcircled{3} & 5 \\ 8 & 8 & 3 & \textcircled{4} \end{array}\right] \qquad (3\text{-}48)$$

The terminal capacity matrix can be put into Form-1 with changing row 3 by row 4.

$$T = \left[\begin{array}{cc:cc} \textcircled{1} & 5 & 2 & 2 \\ 4 & \textcircled{2} & 2 & 2 \\ \hdashline 8 & 8 & \textcircled{3} & 3 \\ 9 & 11 & 5 & \textcircled{4} \end{array}\right] \qquad (3\text{-}49)$$

Realization of (3-49) can be done with Algorithm-A.

1) $$x = \max\left[(-y+7),(3)\right] \qquad (3\text{-}50)$$

$$y = \max\left[(-x+7),(2)\right] \qquad (3\text{-}51)$$

$x=4$ and $y=3$ are obtained from Fig. 3-12.

2) $x+y=7 > t_{43}=5$ and $x < t_{43}$ then Case-2b, $y < t_{43}$ then Case-2d. Applying the numerical values to (3-24) and (3-28) the following equations are obtained.

$$z = \max\left[(-k+4),(3)\right] \qquad (3\text{-}52)$$

$$k = \max\left[(-z+6),(3)\right] \qquad (3\text{-}53)$$

$z=3$ and $k=4$ are obtained from Fig. 3-13.

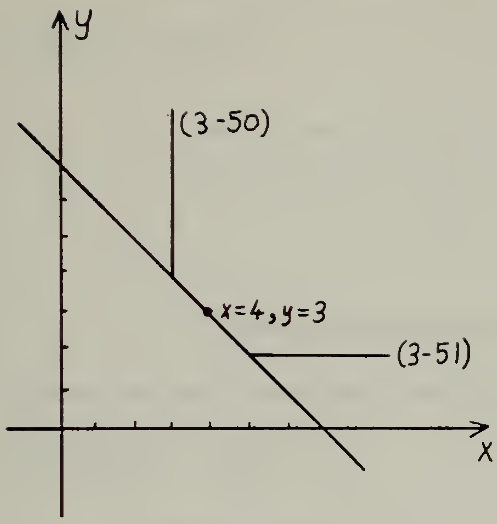


Fig. 3-12

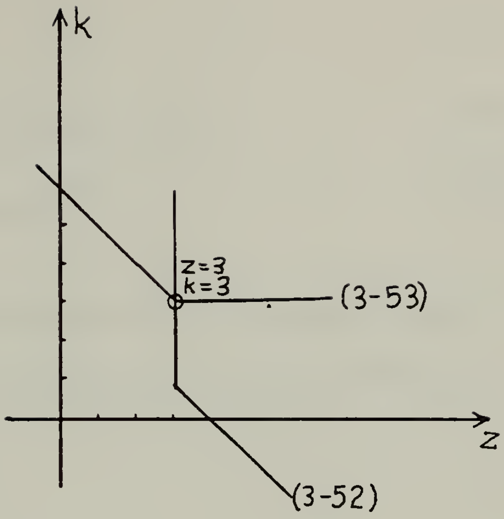


Fig. 3-13

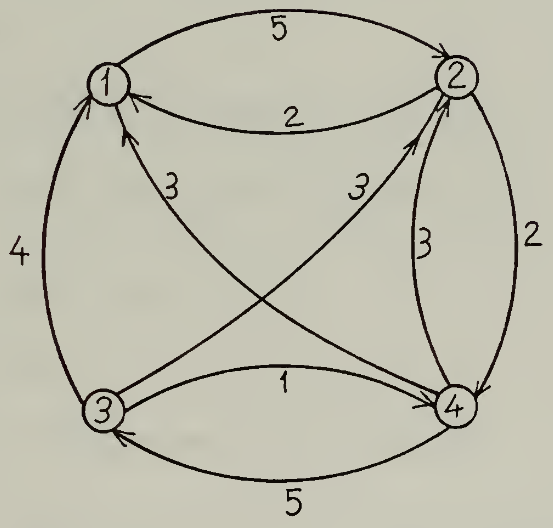


Fig. 3-14. Realization of T matrix in (3-48)

2. Algorithm-B

1) Determine x and z by writing the following equations with the aid of Fig. 3-15:

$$t_{31} = \min\left[(x+z+t'_{21}),\ (x+t'_{21}+t'_{34}+t_1)\right] \tag{3-54}$$

$$t_{41} = \min\left[(x+z+t'_{21}),\ (z+t_{43}+t'_{21})\right] \tag{3-54}$$

Which we obtain

$$x = \max\left[(-z+t_{31}-t'_{21}),\ (t_{31}-t'_{21}-t'_{34}-t_{1})\right] \tag{3-55}$$

$$z = \max\left[(-x+t_{41}-t'_{21}),\ (t_{41}-t_{43}-t'_{21})\right] \tag{3-56}$$

2) There are several sets of equations for determining y and k, choose proper cases and use related equations. Use Fig. 3-16 for obtaining y and k.

CASE-1: $x+z \leq t_{12}$

$$t_{32} = \min\left[(x+y+z+k),\ (x+y+t'_{34})\right] \tag{3-57}$$

$$t_{42} = \min\left[(x+y+z+k),\ (z+k+t_{43})\right] \tag{3-58}$$

from which we get

$$y = \max\left[(-k+t_{32}-x-z),\ (t_{32}-x-t'_{34})\right] \tag{3-59}$$

$$k = \max\left[(-y+t_{42}-x-z),\ (t_{42}-z-t_{43})\right] \tag{3-60}$$

CASE-2: $x+z > t_{12}$

a. $x \geq t_{12}$

$$t_{32} = \min\left[(y+k+t_{12}),\ (y+t_{12}+t'_{34})\right] \tag{3-61}$$

$$y = \max\left[(-k+t_{32}-t_{12}),\ (t_{32}-t_{12}-t'_{34})\right] \tag{3-62}$$

b. $x < t_{12}$

$$t_{32} = \min\left[(y+k+t_{12}),\ (x+y+t'_{34})\right] \tag{3-63}$$

$$y = \max\left[(-k+t_{32}-t_{12}),\ (t_{32}-x-t'_{34})\right] \tag{3-64}$$

c. $z \geq t_{12}$

$$t_{42} = \min\left[(y+k+t_{12}),\ (k+t_{12}+t_{43})\right] \tag{3-65}$$

$$k = \max\left[(-y+t_{42}-t_{12}),\ (t_{42}-t_{12}-t_{43})\right] \tag{3-66}$$

d. $z < t$

$$t_{42} = \min\left[(y+k+t_{12}),\ (z+k+t_{43})\right] \tag{3-67}$$

$$k = \max\left[(-y+t_{42}-t_{12}),\ (t_{42}-z-t_{43})\right] \tag{3-68}$$

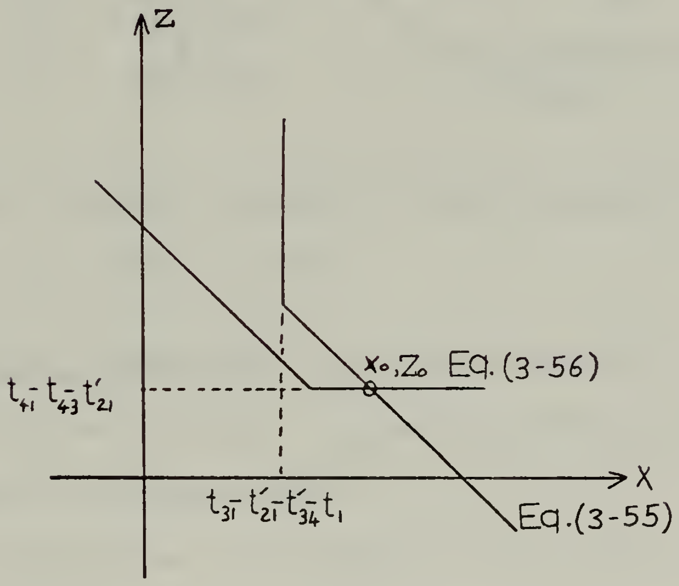


Fig. 3-15. "Curves" of (3-55) and (3-56)

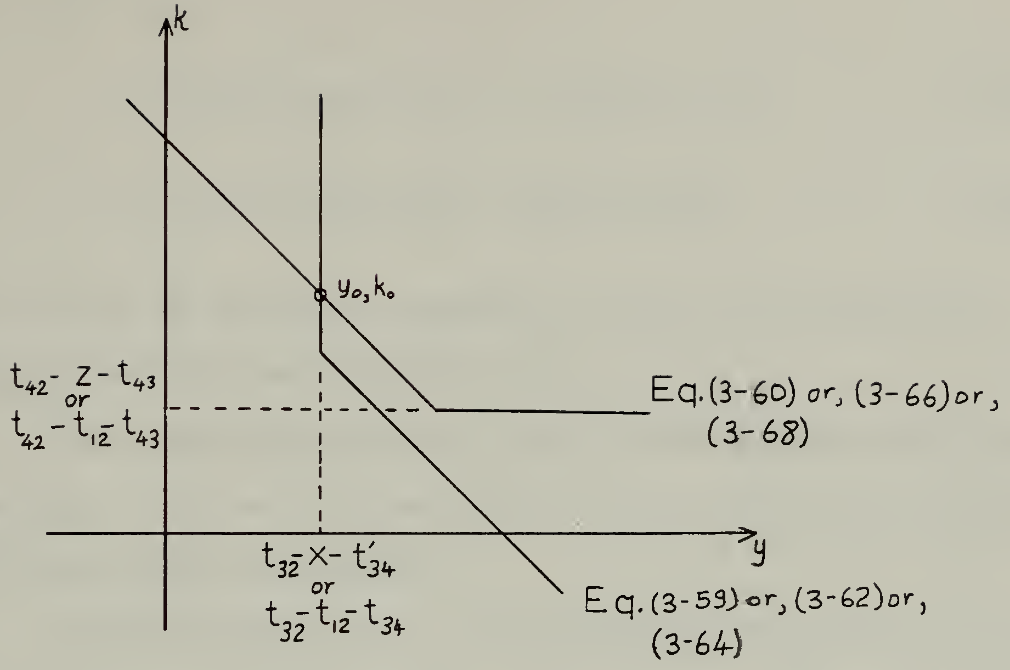

k
y_o, k_o
$t_{42} - z - t_{43}$
or
$t_{42} - t_{12} - t_{43}$
Eq.(3-60) or, (3-66) or, (3-68)
y
$t_{32} - x - t'_{34}$
or
$t_{32} - t_{12} - t_{34}$
Eq.(3-59) or, (3-62) or, (3-64)

Fig. 3-16. "Curves" of (3-59) or (3-62) or (3-64) and (3-60) or (3-66) or (3-68)

Proof of Algorithm-B: Algorithm-B applies in the following cases, $t_{31} < t_{32}$ and t_{42}, $t_{41} < t_{32}$ and t_{42}. From equations (3-29) through (3-32) we can state that:

(a) $x+y+z+k$ can not be min-cut for t_{31} and t_{41} because it is in t_{32} and t_{42}.

(b) $x+y+t'_{34}$ and $z+k+t_{43}$ can not be min-cut for t_{31} and t_{41} respectively, because they are in t_{32} and t_{42} respectively.

Then the following two equations are obtained from (3-29) and (3-31) for t_{31} and t_{41}:

$$t_{31} = \min \left[(x+z+t'_{21}),\ (x+t'_{21}+t'_{34}+t_1) \right] \qquad (3\text{-}69)$$

$$t_{41} = \min \left[(x+z+t'_{21}),\ (z+t_{43}+t'_{21}) \right] \qquad (3\text{-}70)$$

from which we get

$$x = \max\left[(-z+t_{31}-t'_{21}),\ (t_{31}-t'_{21}-t'_{34}-t_1)\right] \tag{3-71}$$

$$z = \max\left[(-x+t_{41}-t'_{21}),\ (t_{41}-t_{43}-t'_{21})\right] \tag{3-72}$$

We can solve the above two equations for x and z or we may use a graphical solution for obtaining x and z.

For the determination of y and k, there are two cases to be considered: (1) $x+z \le t_{12}$ and (2) $x+z > t_{12}$. These cases will give us different sets of equations.

CASE-1. When $x+z \le t_{12}$, automatically $x \le t_{12}$ and $z \le t_{12}$. So $y+x+t'_{34} \le y+t_{12}+t'_{34}$ and $k+z+t_{43} \le k+t_{12}+t_{43}$, then $y+t_{12}+t'_{34}$ and $k+t_{12}+t_{43}$ are eliminated from (3-30) and (3-32) respectively. Also $x+y+z+k \le y+k+t_{12}$, then $y+k+t_{12}$ is eliminated from (3-30) and (3-32) because it can not be min-cut for t_{32} and t_{42}. Thus, (3-30) and (3-32) become

$$t_{32} = \min\left[(x+y+z+k),\ (x+y+t'_{34})\right] \tag{3-73}$$

$$t_{42} = \min\left[(x+y+z+k),\ (z+k+t_{43})\right] \tag{3-74}$$

Which we obtain

$$y = \max\left[(-k+t_{32}-x-z),\ (t_{32}-x-t'_{34})\right] \tag{3-75}$$

$$k = \max\left[(-y+t_{42}-x-z),\ (t_{42}-z-t_{43})\right] \tag{3-76}$$

x and z are found before. In order to obtain y and k we can use a graphical solution using (3-75) and (3-76).

CASE-2. When $x+z > t_{12}$, we can eliminate $x+y+z+k$ from (3-30) and (3-32) since $x+y+z+k > y+k+t_{12}$ and it can not be min-cut for t_{32} and t_{42}. In this case $x \lesseqgtr t_{12}$ and $z \lesseqgtr t_{12}$. Following the same reasoning given in the proof of Algorithm A, equations (3-30) and (3-32) become

$$t_{32} = \min\left[(y+k+t_{12}),\ (y+t_{12}+t'_{34}),\ (y+x+t'_{34})\right] \tag{3-77}$$

$$t_{42} = \min\left[(y+k+t_{12}),\ (k+t_{12}+t_{43}),\ (k+z+t_{43})\right] \tag{3-78}$$

$y+t_{12}+t'_{34}$ or $y+x+t'_{34}$ and $k+t_{12}+t_{43}$ or $k+z+t_{43}$ are eliminated from (3-77) and (3-78) respectively, and each case will give a set of equations for determining z and k.

Example 7. We shall realize the following terminal capacity matrix.

$$T = \left[\begin{array}{cc|cc} \textcircled{1} & 5 & 3 & 3 \\ 4 & \textcircled{2} & 3 & 3 \\ \hline 9 & 10 & \textcircled{3} & 4 \\ 8 & 10 & 3 & \textcircled{4} \end{array}\right] \tag{3-79}$$

The terminal capacity matrix is in Form-2, then realization can be done with Algorithm-B.

1) $$x = \max\left[(-z+8),\ (4)\right] \tag{3-80}$$

$$z = \max\left[(-x+7),\ (4)\right] \tag{3-81}$$

The solution of above equation is obtained from Fig. 3-17, $x=4$ and $z=4$.

2) $x+z=8 > t_{12}=5$, $x < t_{12}$ and $z < t_{12}$ then Case 2b and d. With the numerical values in (3-64) and (3-68), the following equations are obtained:

$$y = \max \left[(-k+5),\ (5) \right] \qquad (3\text{-}82)$$

$$k = \max \left[(-y+5),\ (3) \right] \qquad (3\text{-}83)$$

From Fig. 3-18, y=5 and k=3. Realization of the T matrix is given in Fig. 3-19.

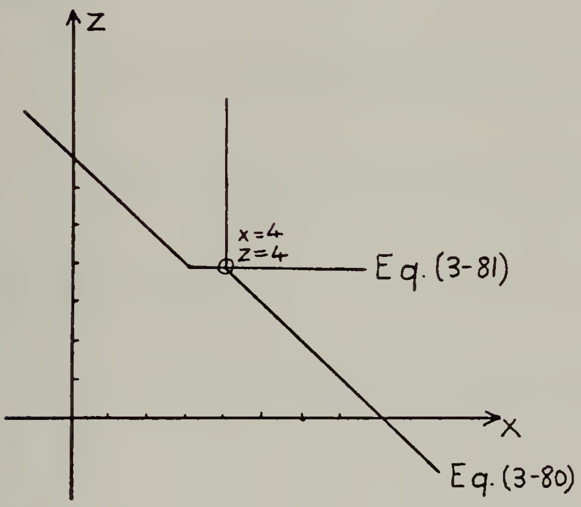


Fig. 3-17

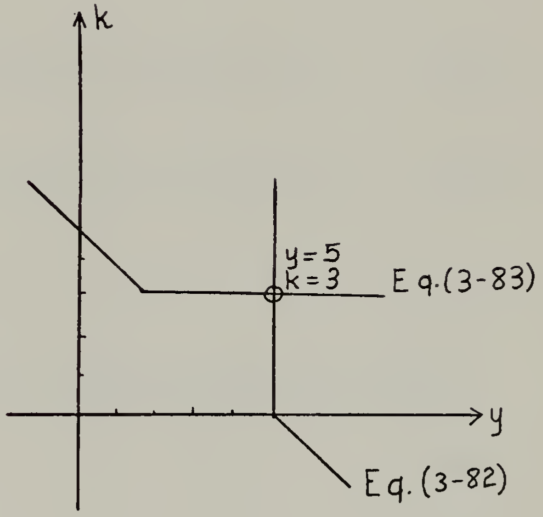


Fig. 3-18

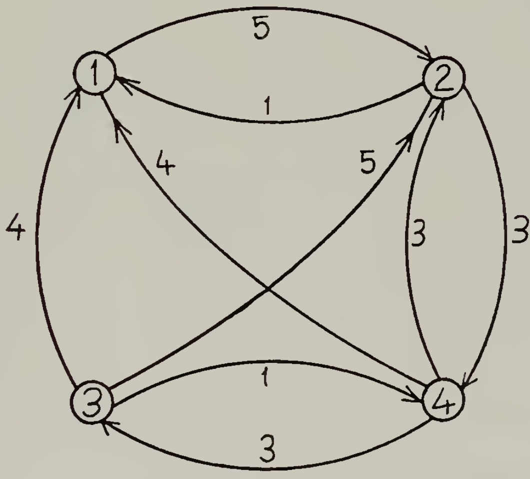


Fig. 3-19. Realization of T-matrix in (3-79)

3. Algorithm C

1) Determine x and k from the following equations.

$$x = t_{31}-t'_{21}-t'_{34}-t_{1} \qquad (3\text{-}84)$$

$$k = t_{42}-t_{43}-t_{12} \qquad (3\text{-}85)$$

2) Determine y and z by writing the following equations with the aid of Fig. 3-20:

$$t_{32} = \min\left[(x+y+z+k),(y+k+t_{12}),(x+y+t'_{34}),(y+t_{12}+t'_{34})\right] \qquad (3\text{-}86)$$

$$t_{41} = \min\left[(x+y+z+k),(z+k+t_{43}),(x+z+t'_{21}),(z+t_{43}+t'_{21})\right] \qquad (3\text{-}87)$$

Which we get

$$y = \max\left[(-z+t_{32}-x-k),(t_{32}-k-t_{12}),(t_{32}-x-t'_{34}),(t_{32}-t_{12}-t'_{34})\right] \qquad (3\text{-}88)$$

$$z = \max\left[(-y+t_{41}-x-k),(t_{41}-k-t_{43}),(t_{41}-x-t'_{21}),(t_{41}-t_{43}-t'_{21})\right] \qquad (3\text{-}89)$$

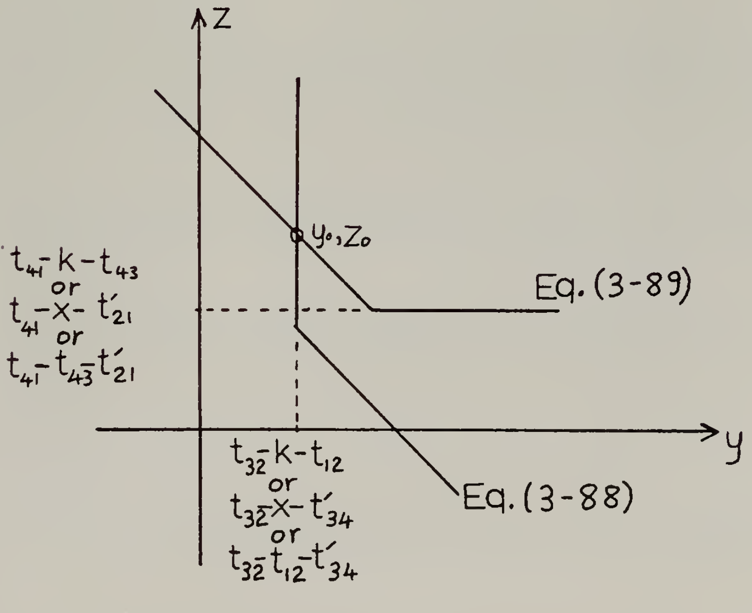

Z
y_0, z_0
Eq. (3-89)
$t_{41}-k-t_{43}$ or $t_{41}-x-t'_{21}$ or $t_{41}-t_{43}-t'_{21}$
y
$t_{32}-k-t_{12}$ or $t_{32}-x-t'_{34}$ or $t_{32}-t_{12}-t'_{34}$
Eq. (3-88)

Fig. 3-20. "Curves" of (3-88) and (3-89)

Proof of Algorithm-C: Algorithm-C applies under the following cases, $t_{31} < t_{32}$ and t_{41}, $t_{42} < t_{32}$ and t_{41}. From equations (3-29) through (3-32) we can state that

(a) $x+y+z+k$ can not be min-cut for t_{31} and t_{42} because it is in t_{32} and t_{41}.

(b) $x+y+t'_{34}$ and $x+z+t'_{21}$ can not be min-cut for t'_{31} because they are in t_{32} and t_{41} respectively.

(c) $y+k+t_{12}$ and $z+k+t_{43}$ can not be min-cut for t_{42} because they are in t_{32} and t_{41} respectively.

Then the following two equations are obtained from (3-29) and (3-32).

$$t_{31} = x+t'_{21}+t'_{34}+t_1 \tag{3-90}$$

$$t_{42} = k+t_{43}+t_{12} \tag{3-91}$$

then

$$x = t_{31}-t'_{21}-t'_{34}-t_1 \tag{3-92}$$

$$k = t_{42}-t_{43}-t_{12} \tag{3-93}$$

For the determination of y and z, the following equations are used.

$$t_{32} = \min\left[(x+y+z+k),(y+k+t_{12}),(x+y+t'_{34}),(y+t_{12}+t'_{34})\right] \tag{3-94}$$

$$t_{41} = \min\left[(x+y+z+k),(z+k+t_{43}),(x+z+t'_{21}),(z+t_{43}+t'_{21})\right] \tag{3-95}$$

Which we obtain

$$y = \max\left[(-z+t_{32}-x-k),(t_{32}-k-t_{12}),(t_{32}-x-t'_{34}),(t_{32}-t_{12}-t'_{34})\right] \tag{3-96}$$

$$z = \max\left[(-y+t_{41}-x-k),(t_{41}-k-t_{43}),(t_{41}-x-t'_{21}),(t_{41}-t_{43}-t'_{21})\right] \tag{3-97}$$

x and k are obtained before, then in (3-96) we have three expressions ($t_{32}-k-t_{12}$, $t_{32}-x-t'_{34}$ and $t_{32}-t_{12}-t'_{34}$) which belong to the constant side of (3-96). Choosing the maximum of the above three expressions two are eliminated. Also in (3-97) choosing the maximum of the constant expressions two are eliminated. After the above simplifications two equations are obtained for determining y and z.

Example 8.: We shall realize the following terminal capacity matrix.

$$T = \left[\begin{array}{cc|cc} \textcircled{1} & 3 & 2 & 2 \\ 5 & \textcircled{2} & 2 & 2 \\ \hline 8 & 11 & \textcircled{3} & 5 \\ 9 & 8 & 4 & \textcircled{4} \end{array}\right] \tag{3-98}$$

The terminal capacity matrix is in Form-3, then the realization can be done using Algorithm-C.

1) $x=0$, $k=1$

2) $$y = \max\left[(-z+10),(7),(8),(4)\right] \tag{3-99}$$

$$z = \max\left[(-y+8),(4),(6),(2)\right] \tag{3-100}$$

which we can write

$$y = \max\left[(-z+10),(8)\right] \tag{3-101}$$

$$z = \max\left[(-y+8),(6)\right] \tag{3-102}$$

From Fig. 3-21 $y=8$ and $z=6$ are obtained. The realization of T matrix is given in Fig. 3-22.

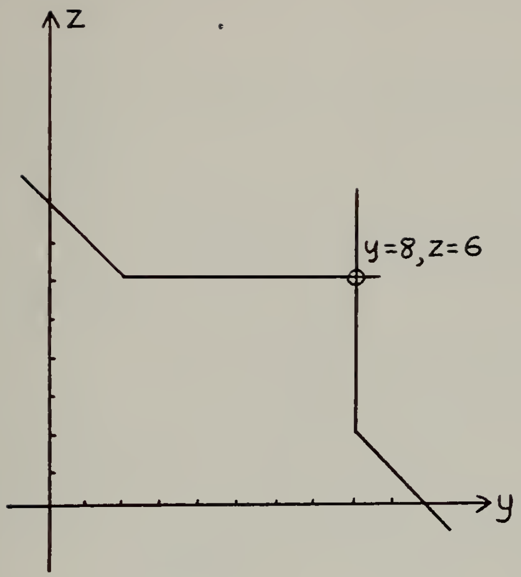


Fig. 3-21

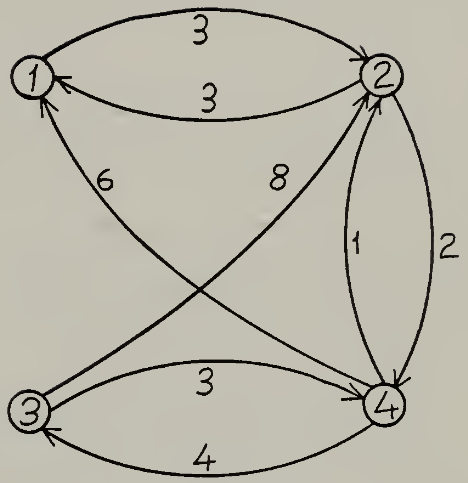


Fig. 3-22. Realization of T matrix

4. Dominant Submatrix Partitioning of T Matrices

The realization technique for the low-order case can be applied to the higher order T matrix if it can be partitioned, by rearranging the nodes, in the following manner:

a) Each submatrix corresponding to a sub-collection of nodes lying along the diagonal line is square.

b) The row of connection node in each diagonal submatrices contains elements with values no smaller than the value of any element in the column of T matrix after treating each diagonal xubmatrix as a node, where each column corresponds to a node which stands for diagonal submatrix. A connection node is a node in each diagonal submatrix which provides connection with the rest of the net.

c) The column of a connection node in each diagonal submatrix contains elements with values no smaller than the value of any element in the row of T matrix after treating each diagonal submatrix as a

node, where the row corresponds to a node which stands for diagonal submatrix.

These conditions are referred to as the "dominant conditions" of a T-matrix.

A T-matrix satisfying the dominant conditions is realizable if [1]:

1) Treating these submatrices along the diagonal line as nodes, the matrix T is realizable.

2) Each submatrix along the diagonal line is realizable.

Example 9: We shall realize the following terminal capacity matrix.

$$T = \left[\begin{array}{c:c:c:c} \begin{matrix}\textcircled{1}&6&3&3\\5&\textcircled{2}&3&3\\10&11&\textcircled{3}&6\\12&13&7&\textcircled{4}\end{matrix} & 1 & 1 & 1 \\ \hdashline 2 & \begin{matrix}\textcircled{5}&5&2&2\\4&\textcircled{6}&2&2\\9&11&\textcircled{7}&8\\8&8&3&\textcircled{8}\end{matrix} & 1 & 1 \\ \hdashline 3 & 5 & \begin{matrix}\textcircled{9}&5&3&3\\4&\textcircled{10}&3&3\\9&10&\textcircled{11}&4\\8&10&3&\textcircled{12}\end{matrix} & 2 \\ \hdashline 5 & 4 & 2 & \begin{matrix}\textcircled{13}&3&2&2\\5&\textcircled{14}&2&2\\8&11&\textcircled{15}&5\\9&8&4&\textcircled{16}\end{matrix} \end{array}\right] \tag{3-103}$$

The T-matrix may be written as

$$T = \left[\begin{array}{cc:cc} \textcircled{A} & 1 & 1 & 1 \\ 2 & \textcircled{B} & 1 & 1 \\ \hdashline 3 & 5 & \textcircled{C} & 2 \\ 5 & 4 & 2 & \textcircled{D} \end{array}\right] \tag{3-104}$$

A, B, C and D are the diagonal submatrices in the original T-matrix.

The realization of the T matrix is shown in Fig. 3-24, where A, B, C and D are the vertices. The realizations of A, B, C and D were done in Example 5 through 8 respectively. The final realization of the T-matrix is shown in Fig. 3-25.

The realization of (3-102) can be obtained using Algorithm-C.

1) $x=0,\ k=1$

2) $$y = \max\left[(-z+4),(4)\right] \tag{3-105}$$

$$z = \max\left[(-y+4),(4)\right] \tag{3-106}$$

From Fig. 3-23 y=4 and z=4 are obtained.

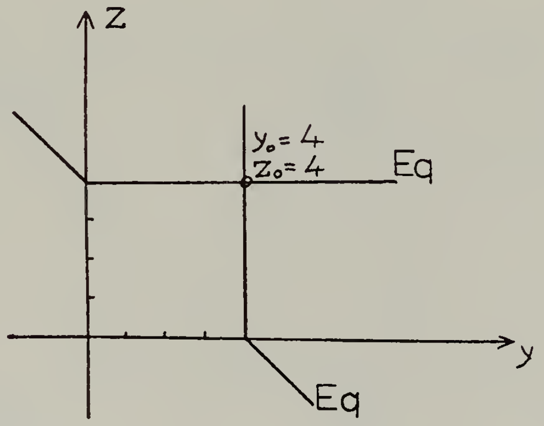


Fig. 3-23

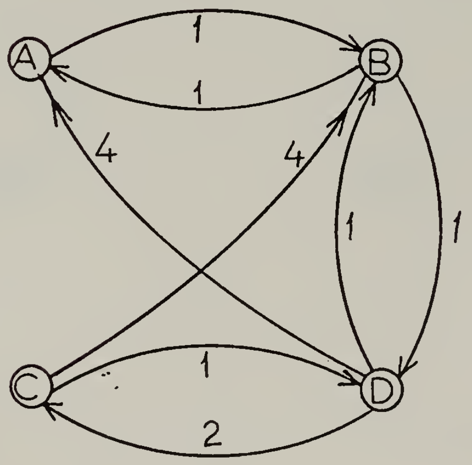


Fig. 3-24

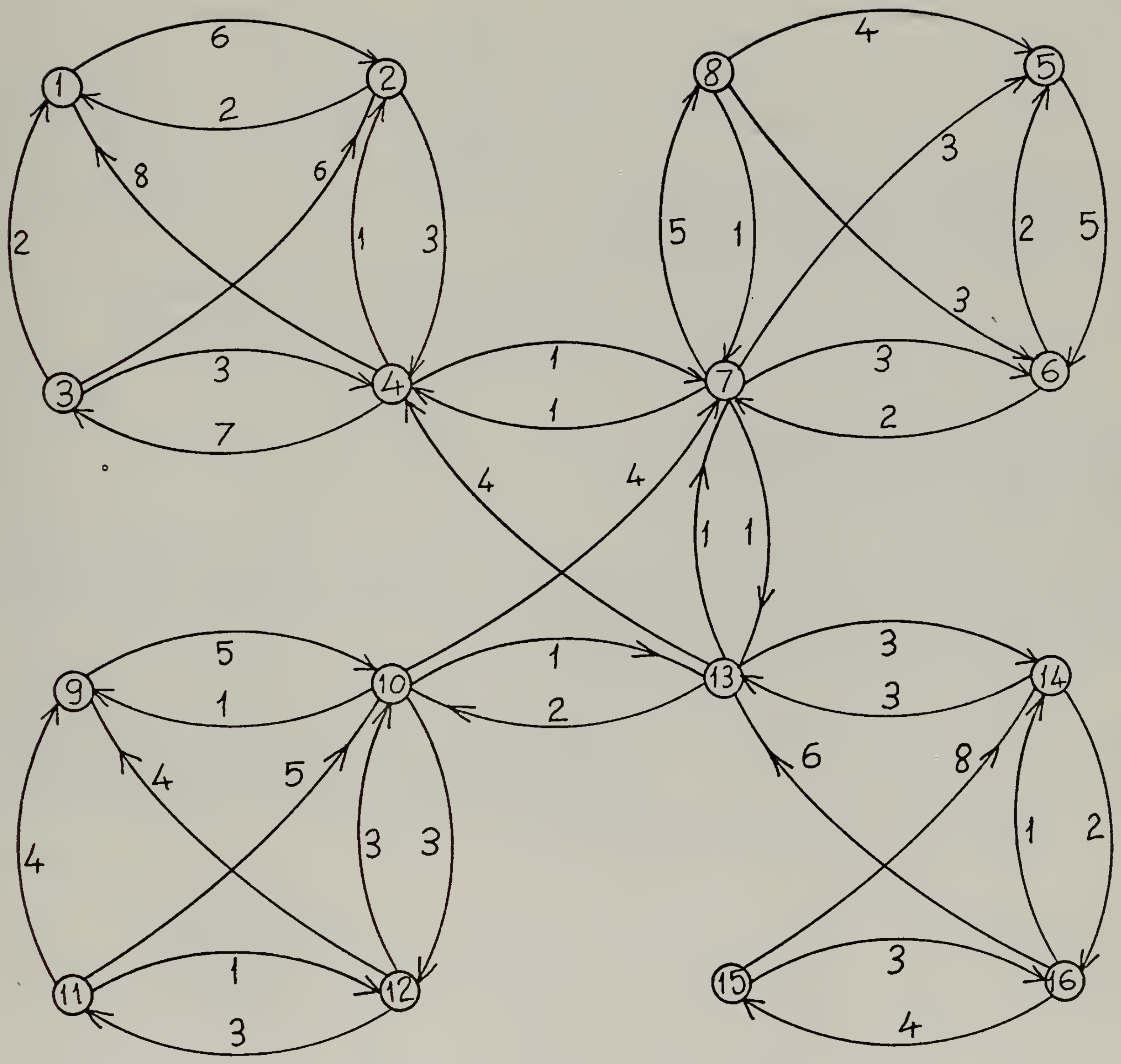
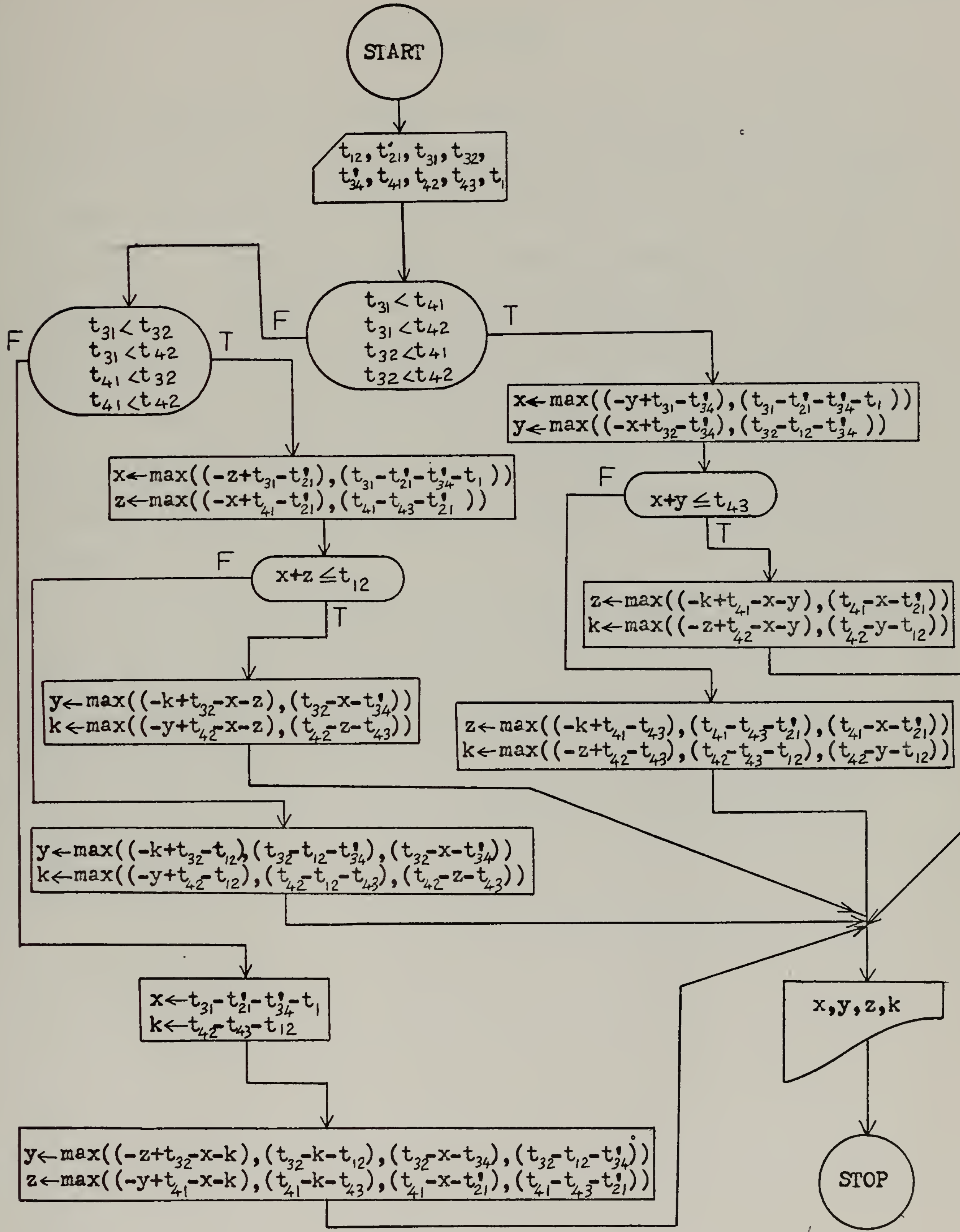


Fig. 3-25. Realization of T-matrix.

5. Flow Chart for Computer Programming


START
$t_{12}, t'_{21}, t_{31}, t_{32}, t'_{34}, t_{41}, t_{42}, t_{43}, t_1$
$t_{31} < t_{41}$ $t_{31} < t_{42}$ $t_{32} < t_{41}$ $t_{32} < t_{42}$
F
T
$t_{31} < t_{32}$ $t_{31} < t_{42}$ $t_{41} < t_{32}$ $t_{41} < t_{42}$
F
T
$x \leftarrow \max((-y+t_{31}-t'_{34}),(t_{31}-t'_{21}-t'_{34}-t_1))$
$y \leftarrow \max((-x+t_{32}-t'_{34}),(t_{32}-t_{12}-t'_{34}))$
F
$x+y \leq t_{43}$
T
$z \leftarrow \max((-k+t_{41}-x-y),(t_{41}-x-t'_{21}))$
$k \leftarrow \max((-z+t_{42}-x-y),(t_{42}-y-t_{12}))$
$z \leftarrow \max((-k+t_{41}-t_{43}),(t_{41}-t_{43}-t'_{21}),(t_{41}-x-t'_{21}))$
$k \leftarrow \max((-z+t_{42}-t_{43}),(t_{42}-t_{43}-t_{12}),(t_{42}-y-t_{12}))$
$x \leftarrow \max((-z+t_{31}-t'_{21}),(t_{31}-t'_{21}-t'_{34}-t_1))$
$z \leftarrow \max((-x+t_{41}-t'_{21}),(t_{41}-t_{43}-t'_{21}))$
F
$x+z \leq t_{12}$
T
$y \leftarrow \max((-k+t_{32}-x-z),(t_{32}-x-t'_{34}))$
$k \leftarrow \max((-y+t_{42}-x-z),(t_{42}-z-t_{43}))$
$y \leftarrow \max((-k+t_{32}-t_{12}),(t_{32}-t_{12}-t'_{34}),(t_{32}-x-t'_{34}))$
$k \leftarrow \max((-y+t_{42}-t_{12}),(t_{42}-t_{12}-t_{43}),(t_{42}-z-t_{43}))$
$x \leftarrow t_{31}-t'_{21}-t'_{34}-t_1$
$k \leftarrow t_{42}-t_{43}-t_{12}$
$y \leftarrow \max((-z+t_{32}-x-k),(t_{32}-k-t_{12}),(t_{32}-x-t_{34}),(t_{32}-t_{12}-t'_{34}))$
$z \leftarrow \max((-y+t_{41}-x-k),(t_{41}-k-t_{43}),(t_{41}-x-t'_{21}),(t_{41}-t_{43}-t'_{21}))$
x,y,z,k
STOP

IV. CONCLUSION

The realization of a terminal capacity matrix with oriented branch capacities is very complicated when the number of nodes becomes large. The techniques given in this paper will be useful for solving more complex communication system problems in practice. The given method can be extended to cover higher order cases without difficulty. The technique presented is effective for the following reasons: (a) The number of nodes in a subnet is increased compared to that in earlier methods. (b) Easy for hand calculation. (c) Adaptable to computer programing. The following problems are suggested for further studies:

1) To obtain a necessary and sufficient condition which is easy to check on a given terminal capacity matrix.

2) To adapt the realization techniques for the nonuniform cost function with minimum cost.

3) Write a computer program for the flow chart presented in this paper.

REFERENCES

1. W. H. Kim and R. T. Chien, Topological Analysis and Synthesis of Communication Networks, Columbia University Press, 1962.

2. D. T. Tang and R. T. Chien, Analysis and Synthesis Techniques of Oriented Communication Nets, IRE Trans. on Circuit Theory, March 1961, p. 39-43.

3. W. Mayeda, Terminal and Branch Capacity Matrices of a Communication Net, IRE Trans. Circuit Theory, September 1960, p. 261-269.

4. S. Seshu and M. B. Reed, Linear Graphs and Electrical Networks, Addison-Wesley Pub. Company, Inc., 1961.

5. O. Wing and R. T. Chien, Optimal Synthesis of a Communication Net, IRE Trans. Circuit Theory, March 1961, p. 44-49.

6. R. E. Gomory and T. C. Hu, Multiterminal Network Flows, J. Soc. Indust. Appl. Math., 1961, p. 551-570.

7. L. R. Ford, Jr. and D. R. Fulkerson, Maximal Flow Through a Network, Canad. J. Math., 1956, p. 399-404.

8. J. A. Resh, On the Synthesis of Oriented Communication Nets, IEEE Trans. Circuit Theory, December 1965, p. 540-546.

9. I. T. Frisch and D. K. Sen, Algorithms to Realize Directed Communication Nets, IEEE Trans. Circuit Theory, December 1967, p. 370-379.

10. W. Chou and H. Frank, Survivable Communication Networks and the Terminal Capacity Matrix, IEEE Trans. Circuit Theory, May 1970, p. 192-197.

11. L. R. Ford Jr. and D. R. Fulkerson, Flows in Networks, United States Air Force Project Rand, August 1962, Princeton University Press.

INITIAL DISTRIBUTION LIST

No. Copies

1. Defense Documentation Center 2
Cameron Station
Alexandria, Virginia 22314

2. Library, Code 0212 2
Naval Postgraduate School
Monterey, California 93940

3. Professor Shu-Gar Chan 2
Department of Electrical Engineering
Naval Postgraduate School
Monterey, California 93940

4. Professor Ralph Bach 1
Department of Electrical Engineering
Naval Postgraduateschool
Monterey, California 93940

5. Lt. Tahsin Karan, TURKISH NAVY 4
Mizrak Sokak
2 Heybeliada, İstanbul
Turkey

6. İstanbul Teknik Universitesi 1
Elektrik Fakültesi
Taşkişla, İstanbul
Turkey

7. Orta-Doğu Teknik Üniversitesi 1
Elektrik Fakültesi
Ankara, Turkey

8. Karadeniz Teknik Üniversitesi 1
Elektrik Fakültesi
Trabzon, Turkey

9. Deniz Kuvvetleri Komutanliği 1
Personel Eğitim Sb. Müdürlüğü
Ankara, Turkey

10. Deniz Harb Okulu Komutanliği 1
Heybeliada, İstanbul
Turkey

11. Deniz Makine Sinif Okullari Komutanliği 1
Derince, Kocaeli
Turkey

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Naval Postgraduate School Monterey, California 93940	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE

On Realization of.Terminal Capacity Matrices

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Master's Thesis; December 1971

5. AUTHOR(S) (First name, middle initial, last name)

Tahsin Karan

6. REPORT DATE December 1971	7a. TOTAL NO. OF PAGES 53	7b. NO. OF REFS 11

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO.	
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Naval Postgraduate School Monterey, California 93940

13. ABSTRACT

This paper presents three algorithms for minimum cost synthesis of an oriented communication net. The realization technique is developed using the min-cut max-flow theorem. The algorithms are able to handle higher order terminal capacities compared to previous methods. Necessary and sufficient conditions are given for the application of the algorithms, which are suitable for computer implementation.

KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Terminal Capacity Matrices						
Terminal Capacity						

21 DEC 72
BINDERY
20325

Thesis
K1426
c.1
133146
Karan
On realization of terminal capacity matrices.
BINDERY
20325
1 DEC 72

Thesis
K1426
c.1
133146
Karan
On realization of terminal capacity matrices.

thesK1426
On realization of terminal capacity matr
3 2768 001 02965 5
DUDLEY KNOX LIBRARY